Feasibility Study of a Pressure Exchanger in an Air-Cycle Air Conditioner

G. W. Herbst, T. A. Newell, and H. H. Korst

ACRC TR-130 August 1997

For additional information:

Air Conditioning and Refrigeration Center
University of Illinois
Mechanical & Industrial Engineering Dept.
1206 West Green Street
Urbana, IL 61801

(217) 333-3115

The Air Conditioning and Refrigeration Center was founded in 1988 with a grant from the estate of Richard W. Kritzer, the founder of Peerless of America Inc. A State of Illinois Technology Challenge Grant helped build the laboratory facilities. The ACRC receives continuing support from the Richard W. Kritzer Endowment and the National Science Foundation. The following organizations have also become sponsors of the Center.

Amana Refrigeration, Inc.
Brazeway, Inc.
Carrier Corporation
Caterpillar, Inc.
Copeland Corporation
Dayton Thermal Products
Delphi Harrison Thermal Systems
Eaton Corporation
Ford Motor Company
Frigidaire Company
General Electric Company
Hydro Aluminum Adrian, Inc.
Indiana Tube Corporation
Lennox International, Inc.
Modine Manufacturing Co.
Peerless of America, Inc.
Redwood Microsystems, Inc.
The Trane Company
Whirlpool Corporation
York International, Inc.

For additional information:

Air Conditioning & Refrigeration Center
Mechanical & Industrial Engineering Dept.
University of Illinois
1206 West Green Street
Urbana IL 61801

217 333 3115

Abstract

The use of a pressure exchanger, or wave rotor, in an air-cycle air conditioner is examined as an alternative to vapor-compression cycles which use chloroflourocarbons (CFC's), hydrochlorofluorocarbons (HCFC's) and hydrofluorocarbons (HFC's). In a pressure exchanger, expansion and normal shock waves are the mechanisms which transfer energy from a high pressure gas, by expanding it, to a low pressure gas, by compressing it. The device consists of a rotor which is divided into cells and a stator end-plate which controls inflow and outflow.

A one-dimensional, homentropic unsteady numerical model was developed using the method of characteristics to simulate the flow through one cell. By entering the inlet and outlet pressures, the locations of the high pressure inlet and outlet ports and the low pressure inlet and outlet ports are calculated based on flow conditions. The mass flow rate through the system and the size of the rotor are also determined.

Besides the pressure exchanger, an air-to-air heat exchanger and air compressor are necessary to develop an air conditioning cycle. These components can be arranged into six configurations of which two are superior to the rest. One configuration consists of the pressure exchanger and compressor working in parallel to raise room air to a higher pressure and temperature so that the heat can be transferred outside. In the other configuration, room air is compressed in stages, first by the pressure exchanger and then by the compressor with heat exchangers located after each compression stage.

Six configurations of these components can provide cooling. Three involve expanding outside air below atmospheric pressure to absorb heat from the cooled space. These are not practical because of variability due to outside ambient humidity, high outlet temperatures for cool air to prevent frosting, and high volumetric flow rates of air. Of the three systems which compress room air to extract energy, the configuration with the pressure exchanger and compressor operating in parallel exhibited slightly better performance than the two configurations where the pressure exchanger and compressor compress the air in series.

Table of Contents

List of Tables

List of Figures

Chapter 1 Problem Statement

Chloroflourocarbons (CFC's, i.e. R-12) have been widely used as the working fluid in air conditioners, refrigerators and other vapor-compression cycles. Studies have found that CFC's leak out of systems over time, rise through the atmosphere and damage the ozone layer. In 1987, the nations of the world signed the Montreal Protocol, which called for the end of worldwide CFC production by 1996. As an temporary solution, hydrochlorofluorocarbons (HCFC's, i.e. R-22) and hydrofluorocarbons (HFC's, i.e. R-134a) have replaced CFC's. But because many of the replacements are greenhouse gases, there is interest finding alternatives to CFC's and HCFC's.

Another approach is the development of environmentally friendly cooling systems which replace the vapor-compression cycle. Some of systems being explored utilize acoustics, desiccants and pressure waves to cool air. Research is being performed at Los Alamos National Laboratory and the Naval Post Graduate School on an acoustic method for refrigeration (Figure 1.1). The system consists of a sealed two-foot long, two-inch diameter pipe filled with 20 atm of helium gas. Two loud speakers, one at each end, resonate at 300 Hz and 180° out of phase creating a wave which drives energy to one end the pipe. Heat exchangers at both ends supply an energy source and sink. Currently, the efficiency is about half of a typical refrigerator.[1]

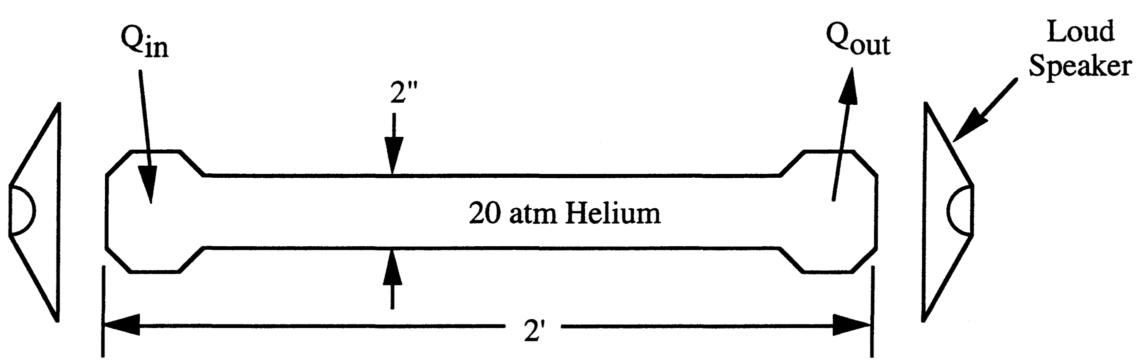
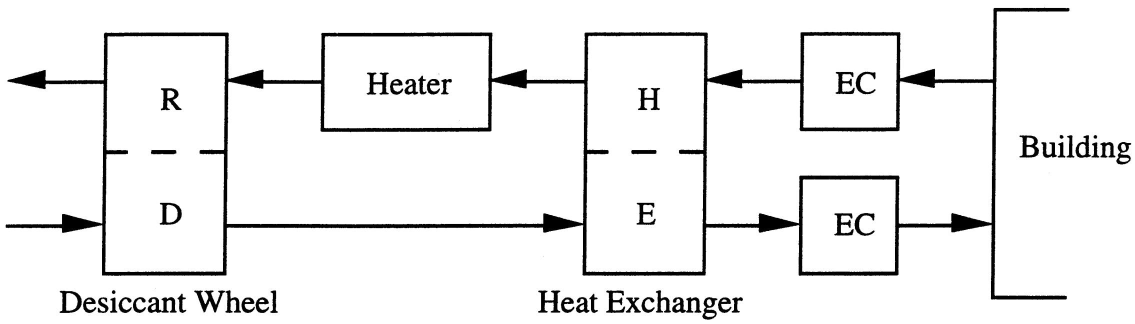


Figure 1.1 Acoustic Refrigeration

Other researchers are examining the integration of desiccants in evaporative cooling systems. Evaporative cooling occurs when relatively dry air passes through a water saturated medium or water spray. Energy from the air evaporates some of the water resulting in a lower air temperature. In the past, evaporative cooling has only been successful in hot, dry climates, but the incorporation of desiccants into the system enables their use in other climates. In this system, the inlet air stream (Figure 1.2) passes through a rotating desiccant wheel. Moisture in the air is absorbed by the desiccant causing the air temperature to increase. The air is then cooled in two steps: 1) sensible energy is removed by a heat exchanger and 2) latent energy is lowered through evaporative cooling, before the air enters the air conditioned space. On the return side, exhaust air from the cooled space passes through an evaporative cooler. The air then passes through the heat exchanger and absorbs the sensible energy from the inlet air stream. Next, the exhaust air is heated, which reduces the relative humidity, so that the air will regenerate the desiccant when it passes through the wheel on its way outside.[2]



where EC - Evaporative Cooler
D - Dehumidification
E - Sensible Cooling
H - Heating
R - Regeneration

Figure 1.2 Desiccant-Based Evaporative Cooling Cycle

In a third system normal shocks and expansion waves transfer energy between two gas streams in a device called a pressure exchanger or wave rotor. Energy is released from an expanding high-pressure gas and the used to compress a low-pressure gas. As an example of a possible air conditioning system (Figure 1.3), air is compressed with the pressure exchanger to a higher temperature and pressure, with a compressor supplying additional compressed air.

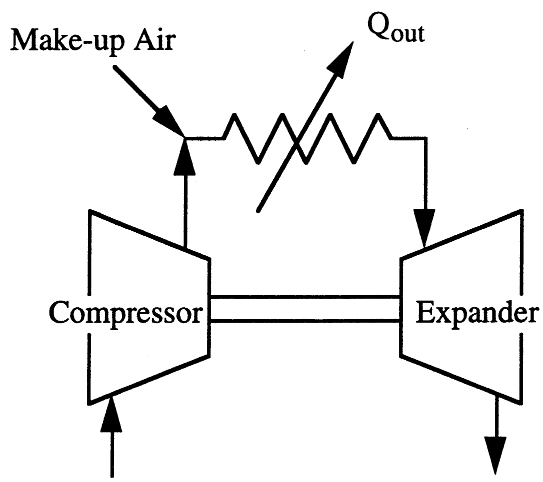


Figure 1.3 Pressure Exchanger Based Air Cooler

A heat exchanger cools the pressurized air. Then the high pressure air is expanded in the pressure exchanger to a cooler outlet temperature, transferring its energy to compress the incoming air. Experimental results by Dr. Kruse at the University of Hanover has shown COP = 1.2[3], where COP is the ratio of energy removed from the system to the work performed on the system.

While Dr. Kruse has taken an experimental approach, this paper takes a theoretical look into the feasibility of using a pressure exchanger in an air conditioning unit. The objectives are:

- Develop a one-dimensional unsteady flow model of the pressure exchanger
- Examine different air conditioning component configurations
- Determine theoretical performance of each configuration

Further development can be concentrated on those configutations which exhibit better performance.

Chapter 2 Pressure Exchanger

2.0) Introduction

The pressure exchanger is a simple device which consists of two basic components: a rotor and a stator with ports (Figure 2.1). The cylindrical rotor is divided axially into cells of constant cross-sectional area. Typically, the rotor is divided into approximately twenty ducts, or eighteen degrees of one revolution. The rotor, which spins on a shaft, can be driven by an external motor connected to the shaft or by the impinging force of the inflow gas on cell walls. Stator plates, located at both ends of the rotor, contain ports which allow gas to enter or leave a cell. Two ports are cut into each stator and ducts are connected to the ports. The four ports consist of: high pressure inlet, high pressure outlet, low pressure inlet and low pressure outlet.

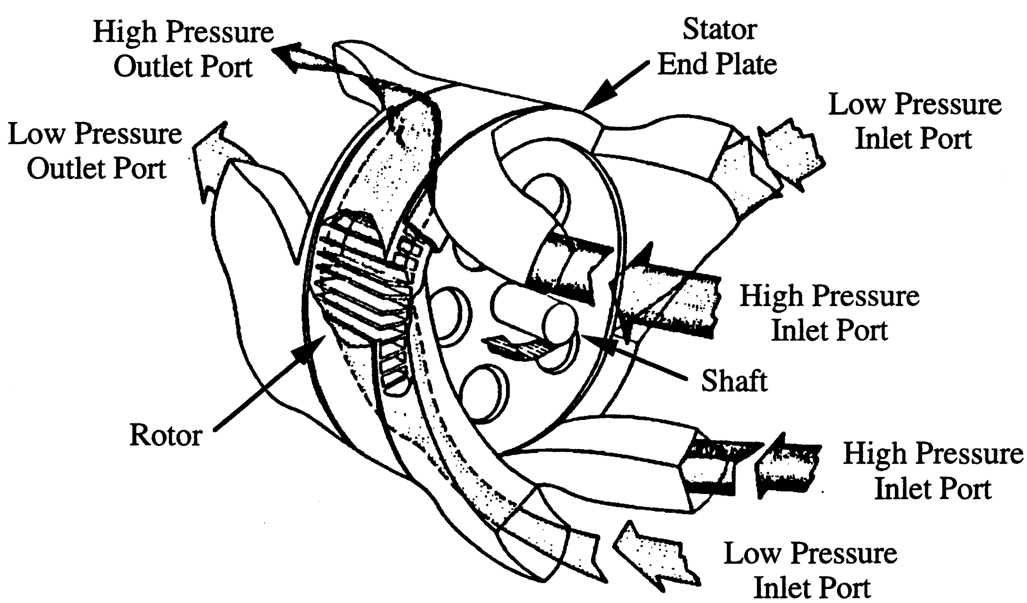


Figure 2.1 Pressure Exchanger Schematic

2.1) Pressure Exchanger Operation

To better understand the compression and expansion mechanisms at work in a pressure exchanger, a frame of reference on the rotor will be used. The cycle can be divided into a high pressure and a low pressure scavenges. Inflow occurs on the left end of the duct with outflow on the right.

With the high pressure scavenge, the shock tube is initially charged with air that is at a pressure between a high pressure reservoir on the left and a low pressure reservoir on the right;

both the inflow and outflow ports are closed (Figure 2.2a). The high pressure inflow port opens which creates a pressure discontinuity between the gas outside the duct (driver) and that inside the duct (driven). As a result, a normal shock develops and propagates rapidly to the right of the duct, raising the pressure of the driven gas. The contact surface between the two gases follows the normal shock to the right at the (slower) velocity of the fluid (Figure 2.2b).

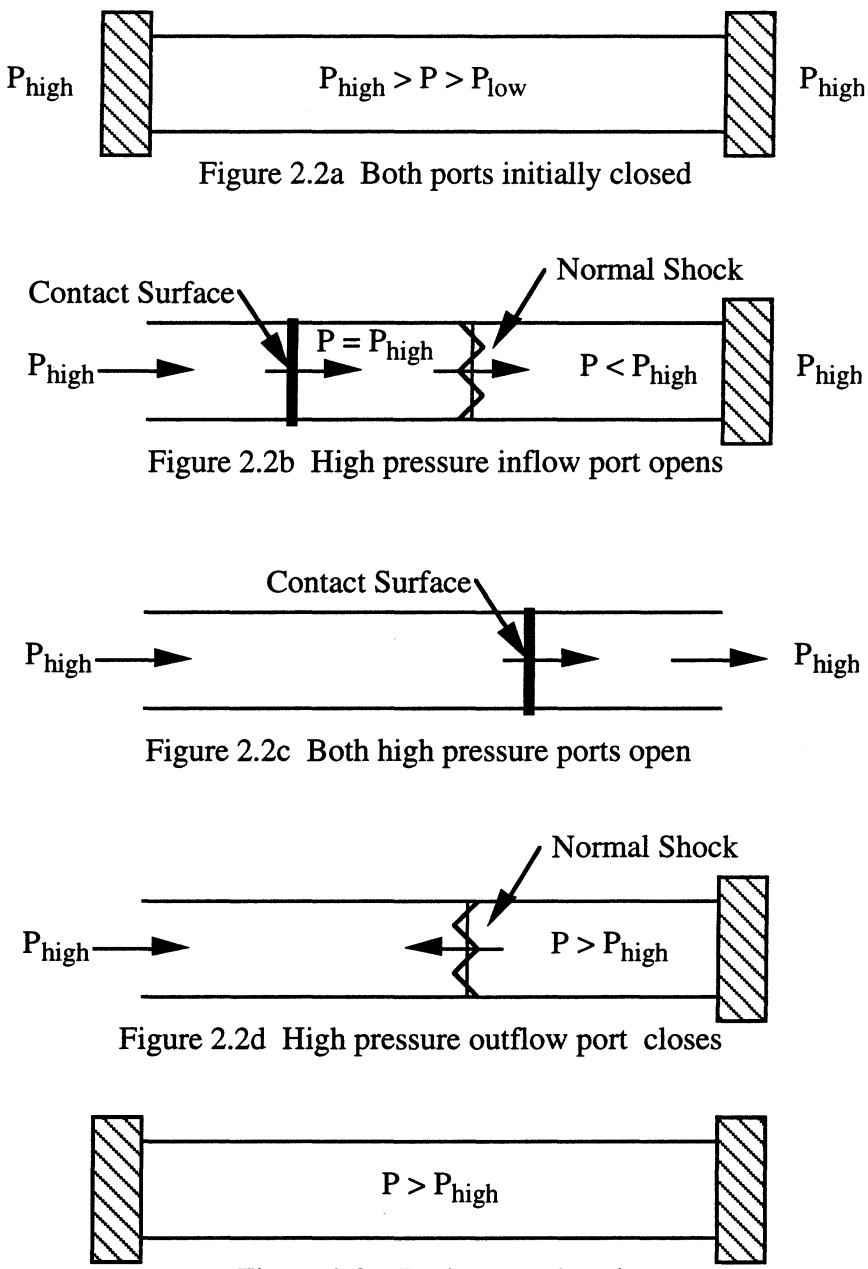


Figure 2.2a Both ports initially closed

Figure 2.2b High pressure inflow port opens

Figure 2.2c Both high pressure ports open

Figure 2.2d High pressure outflow port closes

Figure 2.2e Both ports closed

Figure 2.2 High Pressure Scavenge

When the normal shock reaches the right end and reflects off the closed end, the pressure inside the duct becomes greater than the desired outflow high pressure. The high pressure outflow port opens allowing the pressurized driven gas to exit (Figure 2.2c). The outflow port closes when one of two conditions occur with the contact surface. 1) The contact surface reaches the right end and all the driven gas has exited the duct. 2) The flow at the outlet reverses direction; therefore the maximum driven mass outflow is reached (Figure 2.2d). The high pressure inflow port remains open until the fluid at the inlet reverses direction and begins to flow out. This is the maximum high pressure mass inflow (Figure 2.2e). When both ports have closed, the high pressure scavenge is complete.

The low pressure scavenge is the next process. Initially the duct, with both ends closed, is charged at a pressure between the low and high pressure reservoirs (Figure 2.3a). The process is initiated by opening the low pressure outflow port. Because the pressure inside the duct is higher than outside, a series of expansion waves move upstream causing the driver gas to exit the duct (Figure 2.3b). As the expansion waves reach the left end, the inside duct pressure drops below the outside inlet pressure. At this time the inlet port opens allowing the driven gas in. Both ports remain open as the driver gas exits and the driven gas enters (Figure 2.3c). Once the contact surface between the two fluids reaches the right end, the outflow port closes (Figure 2.3d). The inflow port remains open until the flow at the inlet reverses (Figure 2.3e). Once both ports close, the high pressure scavenge process can begin again.

Combining the two scavenge processes creates the pressure exchanger cycle in Figure 2.4. The duct is initially filled with low pressure gas. When high pressure inlet port opens, the energy from the driver gas compresses the driven gas. The normal shock reflects off the closed port before the high pressure outflow port opens. Once the contact surface reaches the right end, and the port closes, another normal shock is generated which propagates upstream. The closing of the inflow port creates expansion waves. Once both close, the normal shocks and expansion waves continue to move through the duct. When the low pressure port opens, the driver gas expands and exits at a lower pressure. The low pressure inflow port opens so that the momentum of the driver gas helps draw in the fresh driven gas to be compressed. The closing of both ports generates additional normal shocks and expansion waves.

2.3) Pressure Exchanger Background

At the turn of the century, several people were developing static pressure exchangers which did not use pressure wave effects. During the 1930s, C. Seippel in Switzerland and D. Jendrassikin in Hungary, working concurrently and independently, originated the idea of the dynamic pressure exchanger which used unsteady pressure waves to transfer energy.[4]

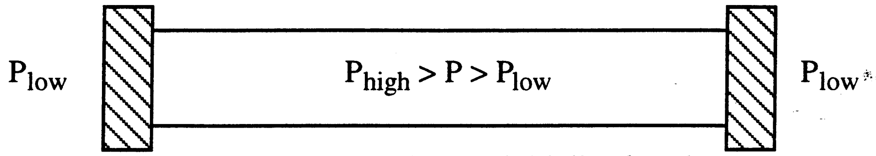


Figure 2.3a Both ports initially closed

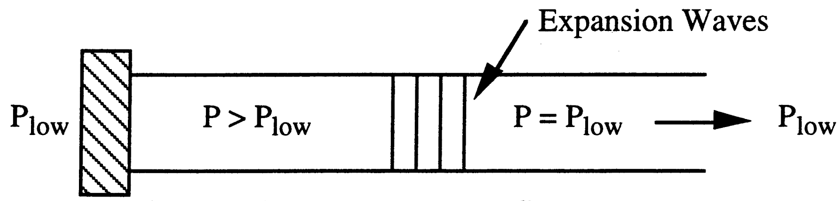


Figure 2.3b Low pressure outflow port opens

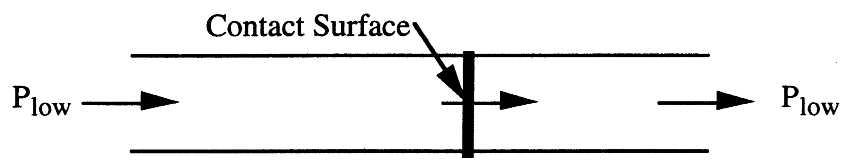


Figure 2.3c Both low pressure ports open

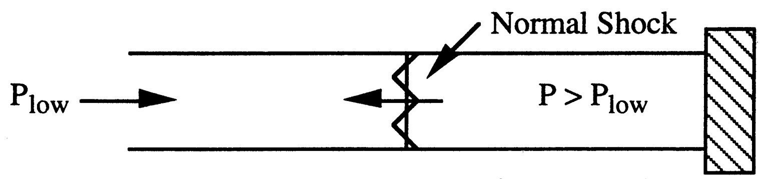


Figure 2.3d Low pressure outflow port closes

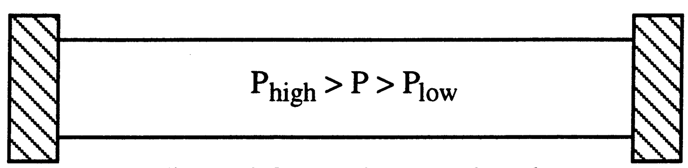


Figure 2.3e Both ports closed

Figure 2.3 Low Pressure Scavenge

One application of this technology is supercharging diesel and gas turbine engines. Development of pressure exchangers was slow until the 1950s when materials were developed which could withstand the cyclical loading experienced during operation. Once this was overcome, most research remained with conventional turbochargers until the 1980s when performance improvements slowed.[5] A major challenge with the pressure exchanger is matching it to engine demand because diesel engines operate over a wide speed range. At slow engine speeds, the normal shock reaches the other end before the port is open, but does not arrive soon enough at high engine speeds. The end plate is designed with ports which recirculate air to the cell at various engine speeds.[6]

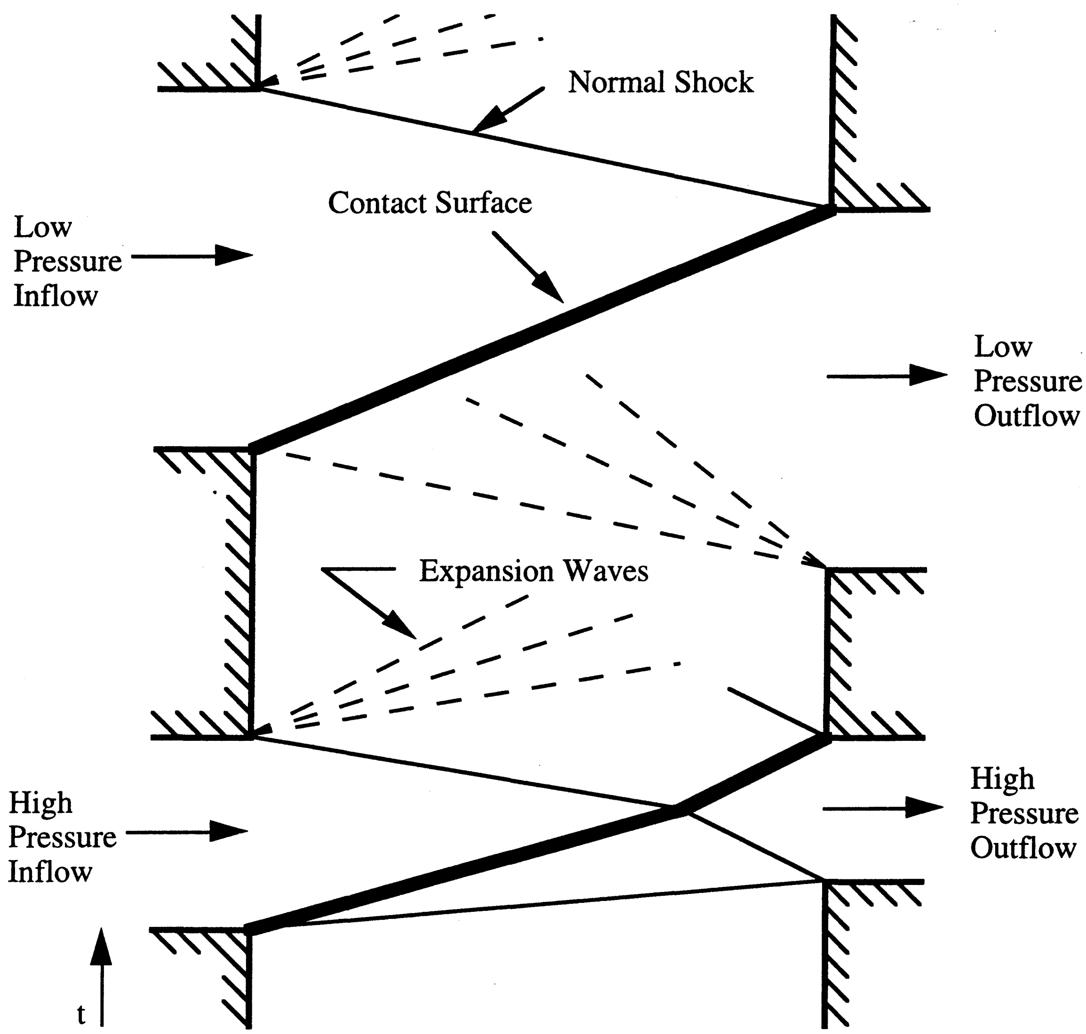


Figure 2.4 Wave Diagram

Other applications including air cooling exist for pressure exchangers. Power Jets (R & D) Ltd. designed and built the first working air conditioner using a pressure exchanger in 1963 for cooling deep mine shafts in South Africa. The pressurized air was cooled with a heat exchanger connected to the mine's water supply. The device used the compressed air system in the tunnels as the source of make-up air. The performance of the first unit was poor with a COP=1.0. A second unit was built in 1965 with a COP = 1.2.[4]

Dr. Kruse at University of Hanover has been the main investigator developing a pressure exchanger based air conditioner. His research team built an experimental unit using parts from engine superchargers. The device achieves a COP = 1.2. Other than Dr. Kruse's work, little work on pressure exchangers for air conditioning have been found.[3]

2.3) Advantages and Disadvantages

Researchers have developed a qualitative list of advantages and disadvantages the pressure exchanger has when compared to turbo-compressors. These include[4]:

Advantages:

- The construction is robust with only one moving part.
- The isentropic efficiencies of compression and expansion are similar to those of turbos.
- Pressure exchangers spin at relatively low rotational speeds compared to turbos.
- The rotor is less prone to erosion due to solid particles or liquid droplets because the fluid velocity is about 1/3 the velocity of gas flowing through a turbo.
- Pressure exchangers can withstand higher temperatures than gas turbines before structural cooling is necessary.
- Pressure exchangers do not exhibit surging performance which many turbos do.
- The pressure exchanger can respond quickly to transients.
- The simple design may result in a lower first cost than comparable turbos.

Disadvantages:

- Mass flow rates per unit of frontal area are lower for the pressure exchanger than a turbo because of the lower fluid velocities.
- Pressure exchangers are not suitable for producing shaft work.
- Pressure exchangers are excessively noisy, a characteristic of unsteady flow.
- Cyclic fatigue loading develops, especially in the rotor, due to the alternating temperatures of the scavenge gases.

While this list of advantages and disadvantages covers most pressure exchanger applications, including superchargers and gas generators where inlet temperatures reach 850°C, several are important for air-cycle air conditioning. The simple, robust design means the unit should provide dependable operation, and the rotor will not be significantly damages by solid particles and water droplets in the air. The excessive sound generation is a disadvantage which will require additional sound suppression to make the system quieter.

Chapter 3 Unsteady Flow Model

3.0) Introduction

A numerical model, which is in Appendix A, was developed to simulate the flow in one cell of a pressure exchanger and to determine the location of ports in the cycle. This computer model was adapted from the program JUNE252I.BAS (see Appendix B) written by Dr. Helmut Korst. The computer model simulates one-dimensional, homentropic flow where the flow properties are calculated by the method of characteristics. Gas Dynamics, Volume 1 written by M. J. Zucrow and J. S. Hoffman[7], are used extensively in developing the method of characteristics portion of the model. Dr. Korst contributed to the program development in addition to offering his program.

3.1) Method of Characteristics

General unsteady flow can be calculated by coupled partial differential equations (pde's). The governing equations for unsteady planar one-dimensional homentropic flow are:

Continuity: $$\rho_t + u\rho_x + \rho u_x = 0 \tag{3.1}$$

Momentum: $$\rho u_t + \rho u u_x + P_x = 0 \tag{3.2}$$

Speed of Sound: $$\left(P_t + uP_x\right) - a^2\left(\rho_t + u\rho_x\right) = 0 \tag{3.3}$$

where ρ is density, u is velocity, P is pressure, and a is the speed of sound. The subscripts designate the partial derivative with respect to time (t) or position (x). Equations 3.1 and 3.3 can be combined to form:

$$P_t + uP_x + a^2\rho u_x = 0 \tag{3.4}$$

For homentropic flow, which will be discussed later, density becomes a function of pressure:

$$\frac{P}{\rho^\gamma} = \text{constant} \tag{3.5}$$

For the two coupled pde's, the independent variables are x and t and the dependent variables are u and P.

Even with only two equations, coupled pde's can be difficult to solve. Along specific curves, called characteristics, the pde's simplify to total differential equations. The two characteristic equations for unsteady 1-D homentropic flow are:

$$d\lambda_{\pm} = \left(\frac{dt}{dx}\right)_{\pm} = \frac{1}{u \pm P^{\frac{\gamma+1}{2\gamma}}} \qquad (3.6)\ \&\ (3.7)$$

where λ is the characteristic, the "+" is a right-running characteristic and the "-" is a left-running characteristic. The total differential equations, which only applies along the characteristic curves, are called the compatibility equations:

$$dP_{\pm} \pm \gamma P^{\frac{\gamma+1}{2\gamma}} du_{\pm} = 0 \qquad (3.8)\ \&\ (3.9)$$

Direct marching or indirect marching are two techniques that may be used to calculate the characteristics. Both methods use an initial value line to begin the calculations. In the direct marching method (Figure 3.1a), left-running (λ_-) and right-running (λ_+) characteristics are projected from the data points (1 and 2) on the initial value. The intersection of the two characteristics is the solution point (3). This method is easy to perform, but the solution point location depends on flow conditions. In the inverse marching method (Figure 3.1b), the location of the solution point (4) is determined.

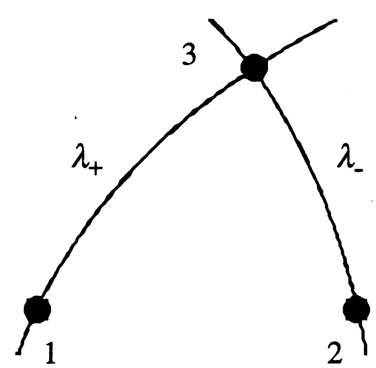


Figure 3.1a Direct Marching Method

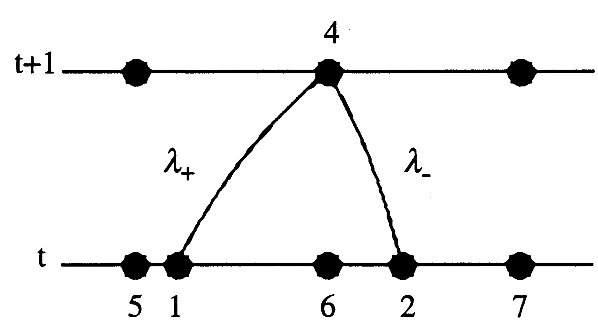


Figure 3.1b Indirect Marching Method

Figure 3.1 Method of Charactericstics

The characteristics are projected back from this location to the initial value line where the properties are interpolated (5, 6 and 7). An updated solution point is calculated with the compatibility equations and the characteristics are again projected back to the initial value line (1 and 2). This iterative process continues until the solution converges. The inverse marching method, while more difficult to implement, is used in this model so that the fluid properties are calculated at a specified locations and time steps throughout the cycle.

The iterations are performed using the predictor-corrector methodology. The solution and characteristic values are predicted based on values from the initial value line. These predicted values are used in the inverse marching technique to calculate corrected values. The corrected values then become the predicted values and the procedure repeats until the predicted and corrected values converge to the solution. Predictor-corrector is second-order accurate.

3.2) Assumptions

This model assumes irrotational, one-dimensional, homentropic flow of an ideal gas through a duct with no friction, heat transfer, or mass transfer. Because the duct has a constant cross sectional area and the length of the duct is greater than its width, one-dimensional flow is valid. Rotational and two-dimensional flow will develop while the ports are opening and closing, but the time scale is short relative to the overall cycle time; therefore, the one-dimensional flow still applies.

The working fluid is also assumed to be homentropic. Homentropy is the state in which the fluid has the same entropy everywhere. The main reason, for this assumption, is to greatly simplify the model. Without the homentropic assumption, density becomes a dependent variable which requires a third characteristic equation to determine the solution point for general 1-D unsteady flow.

It should be noted that normal shocks are treated as isentropic. While this is correct for sonic waves, entropy generation increases across a normal shock as its strength (pressure ratio) increases[8]. For example, a normal shock with a compression ratio of 2 has an isentropic compression efficiency of 94.6% (Figure 3.2). While normal shocks change the entropy within the pressure exchanger, the heat exchanger in the air conditioning cycle causes the entropy to change externally from the pressure exchanger. Because the gas in the system flows through a heat exchanger between compression and expansion in the pressure exchanger, the driver and driven gases will enter the pressure exchanger at different entropies which violates homentropy. The system will be explained in more detail in chapter 4.

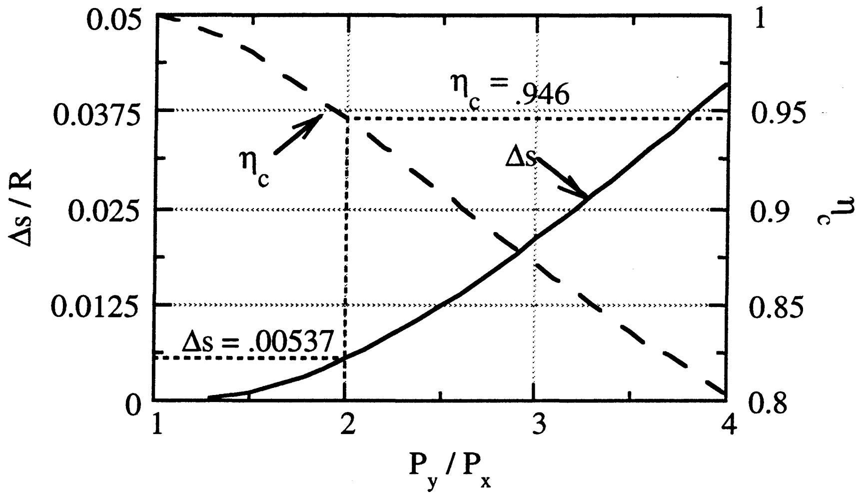


Figure 3.2 Entropy Generation and Isentopic Compression Efficiency Across a Normal Shock

The working fluid in the pressure exchanger is assumed to be an ideal gas. Therefore, no moisture is present. This is only significant if condensation occurs in a cell. Condensation would be modeled as a mass transfer from the cell. But with no condensation, mass transfer is modeled. Friction is something to consider in a more complex model, but is not significant for a basic analysis. The cycle occurs over a small amount of time, allowing little time for heat transfer. Finally, no mixing of the driver and driven fluids is assumed which enables the use of a contact surface between two gases. Therefore, supersonic flow should be avoided.

3.3) Flow

Flow properties at each location in the cell are calculated using one of three numerical adaptations of the method of characteristics: the interior point algorithm, the solid boundary point algorithm or the inflow / outflow point algorithm.

3.3.1) Interior Point Algorithm

The interior point algorithm (Figure 3.3) numerically solves the method of characteristics where the solution node lies between other nodes. The characteristics intersect the previous solution line to the left and right of the solution point. The numerical implementation proceeds as follows:

1. Solution points are given predicted values, designated by the superscript P. The properties of point 4 are set equal to the properties at point 6 ($u_4^P = u_6, P_4^P = P_6$) and the position of point 1 and 2 are equal to 5 and 7, respectively ($x_1^P = x_5, x_2^P = x_7$).
2. The properties at point 1 and 2 are linearly interpolated on the t-solution line. These are the points where the characteristics interect the previous solution line.
3. The speed of sound is calculated at points 1, 2 and 4:

$$a = P^{\frac{\gamma-1}{2\gamma}} \tag{3.10}$$

4. The characteristic equations are solved for x_1^C and x_2^C (corrected values) where the average property value is used:

$$x_1^C = x_6 - dt * \left(\frac{u_1 + u_4}{2} + \frac{a_1 + a_4}{2} \right) \tag{3.11}$$

$$x_2^C = x_6 - dt * \left(\frac{u_2 + u_4}{2} - \frac{a_2 + a_4}{2} \right) \tag{3.12}$$

5. The compatibility equations simultaneously for P_4^C and u_4^C:

$$\left(P_4^C - P_1\right) + \gamma \left(\frac{P_1 + P_4^P}{2} \right)^{\frac{\gamma+1}{2\gamma}} \left(u_4^C - u_1\right) = 0 \tag{3.13}$$

$$\left(P_4^C - P_2\right) - \gamma \left(\frac{P_2 + P_4^P}{2} \right)^{\frac{\gamma+1}{2\gamma}} \left(u_4^C - u_2\right) = 0 \tag{3.14}$$

6. The convergence between the predicted and corrected values of x_1, x_2, P_4 and u_4 are checked. When all four properties converge, the properties at the next time step are known. If solution has not converged, the predicted values are updated with the corrected values and steps 2 through 5 repeat.

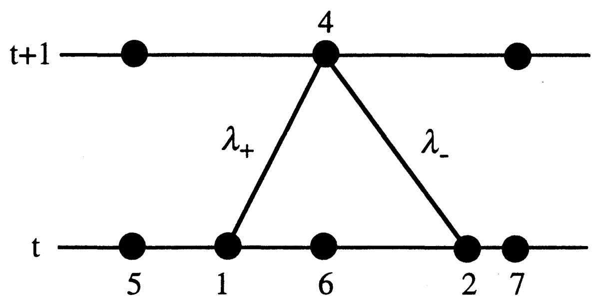


Figure 3.3 Interior Point Algorithm

3.3.2) Solid Boundary Point Algorithm

With the solid boundary point algorithm (Figure 3.4), the solution point is located on a surface. As a result, either the left-running or the right-running characteristic falls outside of the system. This requires an addition equation to solve the system because only one characteristic is usable. According to boundary layer theory, a point adjacent to a surface has the same velocity as that surface. When the ports are closed, $u_4 = 0$.

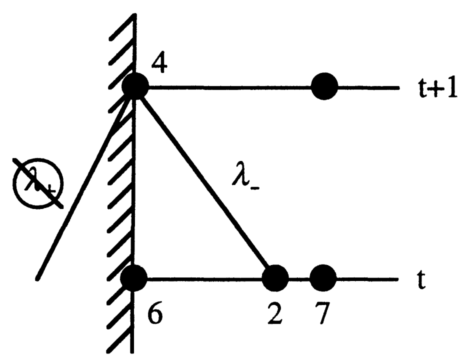


Figure 3.4 Solid Boundary Point Algorithm

3.3.3) Inflow / Outflow Point Algorithm

With the inflow/outflow case (Figure 3.5), fluid is entering or leaving the system. Like the solid boundary point, only one characteristic intersects the previous solution line for subsonic flow; the other falls outside the system. While the surface velocity condition may be used in boundary point algorithm, the velocity through an open port is not known. For subsonic flow, the pressures in the cell and duct must be equal which constrains the system. This condition will be examined further in the next section.

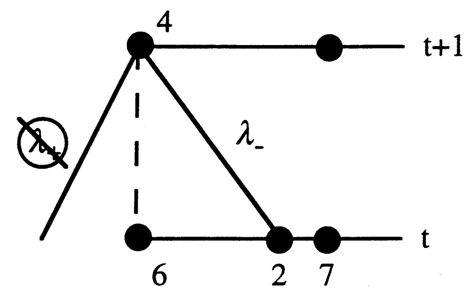


Figure 3.5 Inflow / Outflow Point Algorithm

3.4) Opening and Closing Ports

As stated earlier, the typical pressure exchanger is divided into approximately twenty cells or eighteen degrees of rotation. The program takes time steps between one and two degrees. As a result, an area restriction exists at the ports while they are opening or closing. Therefore, the ports are modeled as gradually opening and closing.

3.4.1) Inflow Through a Partially Open Port

When gas flows through a partially open port, the flow accelerates in the duct as it approaches the port. The flow reaches its maximum velocity at the port, where the area restriction occurs. Once inside the cell, the flow experiences a sudden expansion as it enters a cell with constant cross-sectional area. This process can be modeled as two pieces: 1) convergence in the duct and 2) sudden expansion in the cell.

The acceleration in the duct up to the port can be modeled as isentropic flow through a short converging where the port is the throat of the nozzle. Isentropic flow equations calculate the pressure based on Mach number (velocity and speed of sound)[8]:

$$\frac{P}{P_0} = \left(1 + \frac{\gamma + 1}{2} M^2\right)^{\frac{\gamma}{\gamma+1}} \tag{3.15}$$

A sudden enlargement of a duct (Figure 3.6a) best represents the expansion of the gas in the cell. The equations are:

$$\frac{P_2}{P_1} = \frac{A_1}{A_2}\left[\frac{P}{P^*}(\gamma, M_2)\right]\left[\frac{P}{P^*}(\gamma, M_1)\right]^{-1} \tag{3.16}$$

$$\frac{P_{base}}{P_1} = \left(\gamma + 1\left[\frac{A_1/A_2}{1 - A_1/A_2}\right]\right)\left[\frac{F}{F^*}(\gamma, M_2) - \frac{F}{F^*}(\gamma, M_1)\right]\left[\frac{P}{P^*}(\gamma, M_1)\right]^{-1} \tag{3.17}$$

where

$$\frac{P}{P^*}(\gamma, M) = M^{-1}\left[\left(\frac{2}{\gamma+1}\right)\left(1+\frac{\gamma-1}{2}M^2\right)\right]^{-1/2} \tag{3.18}$$

$$\frac{F}{F^*}(\gamma, M) = M^{-1}\left[1+\gamma M^2\right]\left[2(\gamma+1)\left(1+\frac{\gamma-1}{2}M^2\right)\right]^{-1/2} \tag{3.19}$$

where point 1 is at the throat and point 2 is downstream with uniform one-dimensional flow. Either sonic or subsonic can occur in a sudden enlargement. When $P_1 = P_{base}$, the flow in the cell is subsonic. When $P_1 \geq P_{base}$, the flow is sonic at the throat, but the flow becomes subsonic downstream. Flow separation at the nozzle occurs because of the sudden enlargement[9]. While this describes the actual operation, unfortunately a solution does not exist for all conditions; therefore, the partially open inflow port was modeled as an isentropic converging-diverging (c-d) nozzle. (Figure 3.6b)

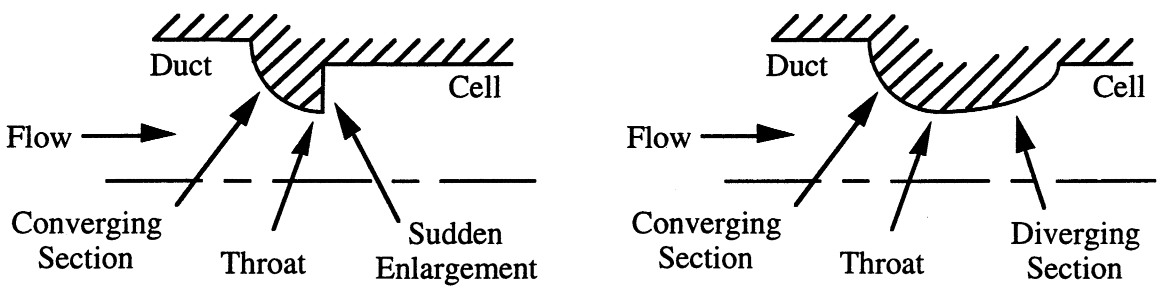


Figure 3.6a Sudden Enlargement

Figure 3.6b Converging-Diverging Nozzle

Figure 3.6 Inflow Through a Partially Open Port

The converging section of a c-d nozzle is the same as with the sudden enlargement. The difference exists with the expansion section. In an ideal diverging nozzle, the cross-sectional area increases gradually to prevent separation. Subsonic flow at the exit is desired at the nozzle exit so that mixing does not occur. Therefore two flow regimes exist where nozzle flow at the exit is subsonic. In the first regime, the throat pressure and cell pressure are equal and flow is subsonic throughout the nozzle. In the other regime, the flow is choked at the throat where it transitions to supersonic flow. A normal shock develops in the diverging section to equilibrate pressures and causes the supersonic flow to become subsonic. Ultimately, the cell pressure becomes so low that the exit flow is supersonic which violates the no mixing assumption and is not allowed. While a c-d nozzle is too long for practical use, a solution can be calculated for all conditions. The converging-diverging nozzle conserves more kinetic energy of the fluid than the sudden enlargement because no separation occurs in an isentropic c-d nozzle.

3.4.2) Outflow Through a Partially Open Port

Outflow is modeled as gas flowing through a isentropic converging nozzle (Figure 3.7). The flow is restricted to subsonic flow inside the nozzle because of the no mixing assumption. The partially open port is acts as the throat of a nozzle which causes the flow to accelerate. The duct pressure will determine if the flow is choked at the exit port. When $P_{duct} \geq .52828 * P_{cell}$, the flow can accelerate so that the $P_{throat} = P_{duct}$. When $P_{duct} \leq .52828 * P_{cell}$, the flow chokes at the port creating sonic flow at the throat. The pressure at the throat and in the duct become independent. The resulting flow in the duct is not important because mixing in this region does not violate the assumption.

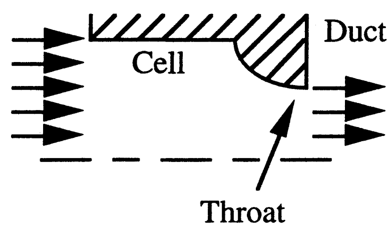


Figure 3.7 Outflow Through a Partially Open Port

3.4.3) Inflow Through a Fully Open Port

Flow entering a completely open port is a special case of the partially open nozzle. It is modeled as a converging nozzle where the throat has the same area as the cell because no sudden enlargement occurs.

3.4.4) Outflow Through a Fully Open Port

Flow exiting a totally open port is also a special case. The gas does not accelerate as it exits the cell because no area restriction exists. Therefore, the cell pressure at the port and the duct pressure are equal for subsonic flow.

3.5) Port Timing

The ports' opening and closing are based on the conditions discussed in Chapter 2. The ouflow ports close when the either the contact surface reaches the end or when the flow reverses direction. The high pressure outflow and low pressure inflow ports open once conditions allow the appropiate outflow and inflow, respectively. The timing of the first port opening which initiate the high and low scavenge processes is not discussed there. One approach is waiting for uniform conditions throughout the cell before opening the port (Figure 3.8). When a scavenge is complete and both ends close, normal shocks and expansion waves continue to propagate through the duct causing the pressure and the velocity to oscillate. Eventually the oscillations dampen to a no flow, constant pressure condition. At this time the first port opens to initiate the next scavenging process. One cycle take 29.8 dimensionless time units and it does not take advantage of the momentum of the fluid.

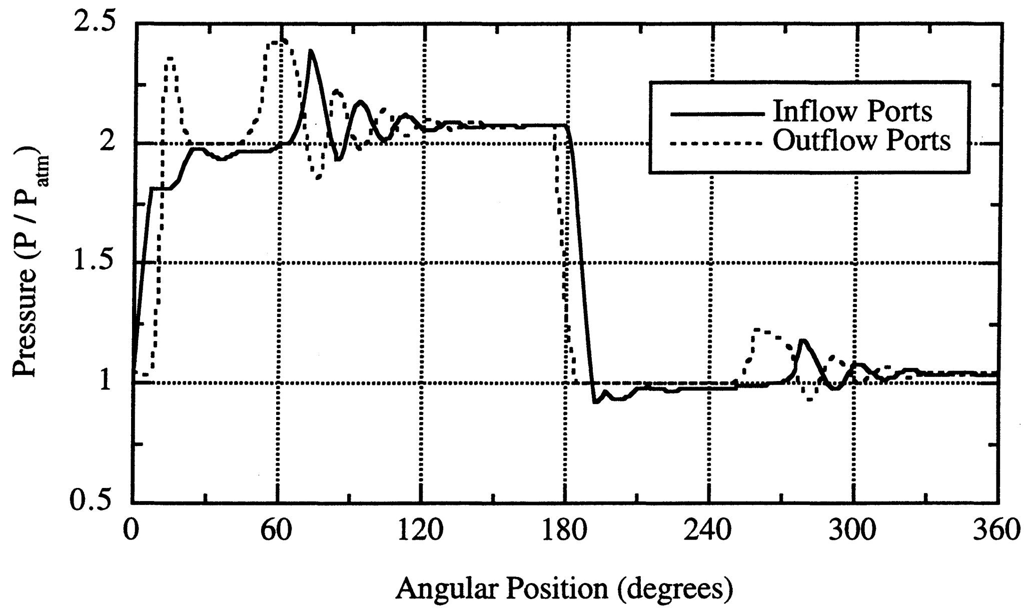


Figure 3.8 Uniform Flow at Scavenge Initiation

With the second option, the port opens when the local velocity reaches a maximum so that the momentum of the oscillating fluid (Figure 3.9) is used. For the high scavenge process, the driven gas will draw the driver gas into the left end of the duct. While at the right end, the driver gas will already be moving toward the outlet port when the low pressure scavenge begins. Also, the overall cycle time 14.7 dimensionless time units is shorter, because the ports are closed for less time, which increases the mass flow rate.

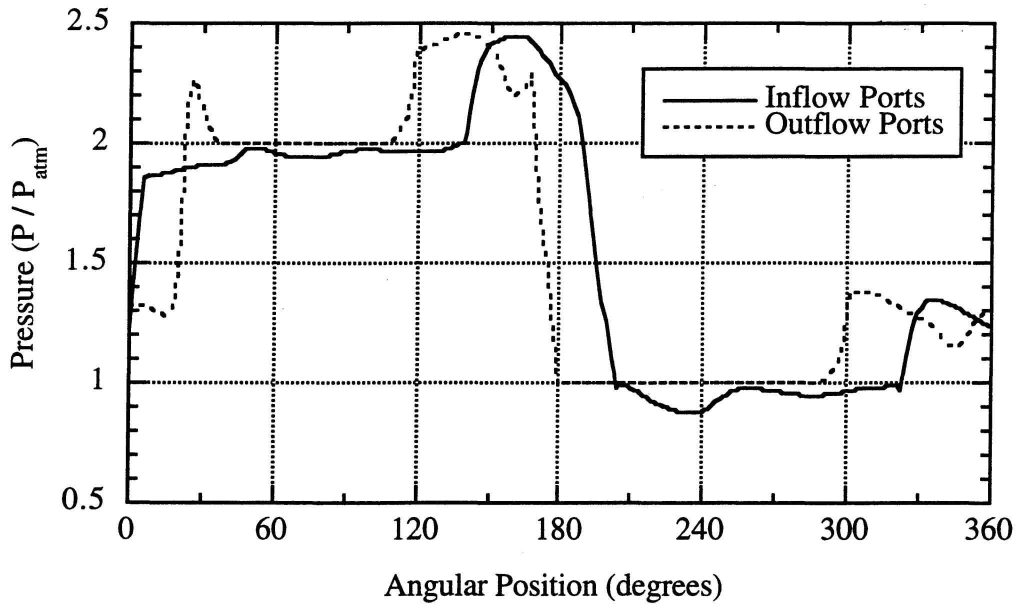


Figure 3.9 Maximum Momentum at Scavenge Initiation

3.6) Contact Surface

The position of the contact surface between the two gases is important because the closing of both outflow ports depend on when all of one gas has left the cell. This is done by interpolating the fluid velocity at the current location of the contact surface. By multiplying this velocity by the change in time, this results in the distance moved.

3.7) Mass Flux

The mass flux into the system is calculated for each time step, which is the product of the local density and velocity of the gas and the size of the time step. The density is a function of pressure and is calculated by equation 3.5. The mass flow rates are summed to determine the total mass flow rate for the cycle.

3.8) Pressure Exchanger Design

The rotor dimensions can be calculated with the flow properties and performance parameters. The length of the rotor is a function of the dimensionless time steps (dt), the angular speed of the rotor (RPM), and the speed out sound at the inlet (a_0):

$$dt = \left(\frac{60}{\mathrm{RPM}}\right)\left(\frac{1}{\#\,\mathrm{Steps/rev}}\right)\left(\frac{a_0}{L}\right) \tag{3.20}$$

The frontal area of the rotor is a function of the dimensionless mass flux (m''), mass flow rate ($\dot{m}$) and inlet pressure of the fluid (P_0):

$$m'' = \left(\frac{\dot{m}}{\dot{V}A}\right)\left(\frac{1}{\rho_0}\right) \tag{3.21}$$

where the mass flow and volumetric flow rates are a known parameters and density is calculated by Equation 3.5.

3.9) Program Operation

This program models one cell as it completes a cycle. Position and time are the independent variables used to calculate pressure and velocity, the dependent variables where all the variables are dimensionless. Data files are used to enter the input variables and save the results.

3.9.1) Input Variables

The program begins by retrieving the operating parameters. The number of stations across the cell and size of each time step establish the grid resolution. The computer is the able to calculate the maximum allowable time step for which the solution will remain stable. Next, the desired pressures in the ducts are entered which will determine the port locations and flow properties. Also, the number of time steps required for the port to fully open or close is read. Finally, a time is entered for each port at which that port will automatically begin to close. These values are overwritten if the flow conditions necessitate the closing of the ports before this time is reached.

3.9.2) Cycle Calculations

With these initial inputs, the program begins by initializing the previous solution line at t=0. Both ports are closed and no flow exists at t=0. The pressure inside the cell is set equal to the average of the four port pressures. At each time step, the program follows this procedure:

At the first time step, t=1, the high pressure inlet port starts opening. The general procedure is:

1. Determine which ports, if any, will open or close. The high pressure inflow port always open at the first time step, t=1.
2. Calculate how much the ports are open.
3. Calculate flow properties at the next time step using method of characteristics starting at the left and moving to the right.
4. Calculate the location of the contact surface in the cell.
5. Determine if the properties just before the high pressure inlet port opens have converged with the intial values. If yes, then the flow properties are calculated.
6. Update the pressure and velocity values at t=0 with the final values.
7. Repeat steps 1 - 5.

When the cycle has reached steady-state conditions, the time steps are converted to angular positions and the output files are created.

3.9.3) Output Variables

The output files contain the pressure and velocity values for every time step and angular position at each station. Also, the angular position where the each port opens and closes is stored. Finally, the position of the contact surface in the cell is recorded.

3.10) Example

The following is an example of the results from computer simulation of a steady-state cycle. Both high pressure ports are set at 202 kPa and the low pressure ports are at 101 kPa. Eleven nodes are equally spaced the length of the duct and the dimensionless time step is .05, the maximum for stability.

3.10.1) Flow Properties

In Figures 3.10 and 3.11, the pressures and velocities, respectively, are plotted for selected locations versus the position in the cycle. With the opening of the high pressure inlet port, normal shocks move downstream raising the pressure and increasing velocities as can be seen from 0° to 30°. When the pressure at the right end is greater than the outlet pressure, the high pressure outflow port opens. The series of normal shocks cause the pressure at the right end to overshoot the target outlet pressure 30° to 45°, but the pressure drops back to the target value once the port is fully open. While both high pressure ports are open 45° to 100°, the pressure and velocity oscillate as weak normal shocks and expansion waves move through the cell.

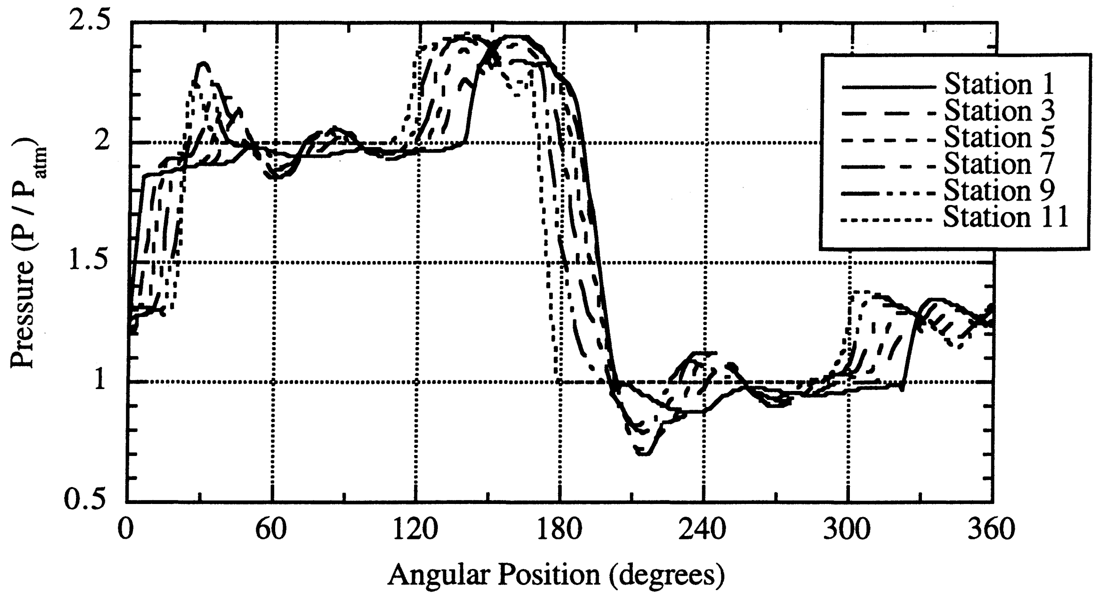


Figure 3.10 Pressure at Stations During Cycle

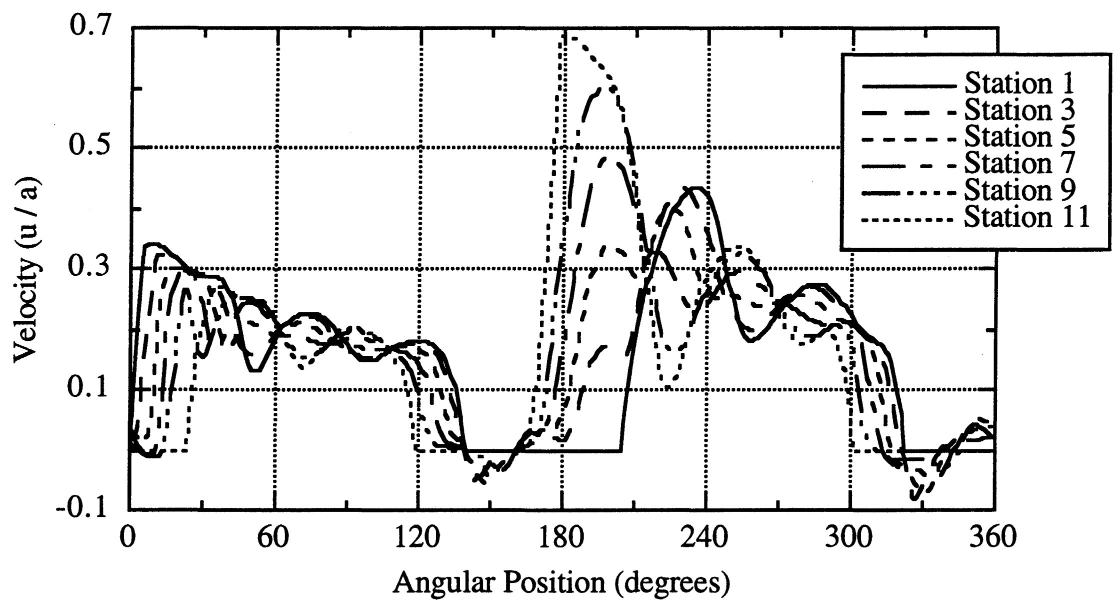


Figure 3.11 Velocity at Stations During Cycle

As the contact surface reaches the right end 100° to 120°, the high pressure port closes sending normal shocks back toward the inlet port which slows the velocity of the gas in the cell. When the low pressure port opens 160° to 190°, a series of expansion waves accelerate the flow to the right while dropping the pressure. Under the proper conditions, the low pressure inflow port opens. The low pressures contact surface moves downstream through oscillating normal shock and expansion waves 200° to 300°. Eventually, both ports close and the cycle is complete.

3.10.2) Port Locations

The ports open and close based on the pressure and velocity conditions just described. The angular position where the ports begin opening and closing is shown in Table 3.1. Each rotor does not see a fully-open or fully-closed port for 10 time steps or 12°. One revolution consists of 140° for the high pressure scavenge, 155° for the low pressure scavenge, and 65° when no ports are open.

Scavenge	Inlet	Outlet
High Pressure	0.0° - 127.8°	23.3° - 106.9°
Low Pressure	204.0° - 309.6°	167.1° - 288.7°

Table 3.1 Pressure Exchanger Port Locations

3.10.3) Contact Surface Movement

The position of the contact surfaces moving across the cell are shown in Figure 3.12. The high pressure scavenge begins at 0° when the high pressure inlet port opens and reaches the other end at 115° as the outlet port finishes closing. The low pressure contact surface develops at 204° when the inlet port opens and reaches the outlet port at 300°C. The contact surface does not move at a constant velocity because fluid velocity fluctuates as normal shocks and expansion waves propagate through the cell.

3.10.4) Mass Flux

The inlet mass flux (Figure 3.13) follows a similar path to the location of contact surface because as the contact surface moves down the cell, more mass is able to enter the system. The high pressure mass flux begins when the high pressure inlet ports opens and continues until it closes. While the contact surface reaches the outlet at 114°, mass continues to flow into the system until 137° because the outlet port closes before the inlet port. Therefore, the cell is charge with more high pressure gas than it releases. The low pressure mass flux is similar with the contact surface completing the scavenge at 300° while mass continues to enter the system until 312°.

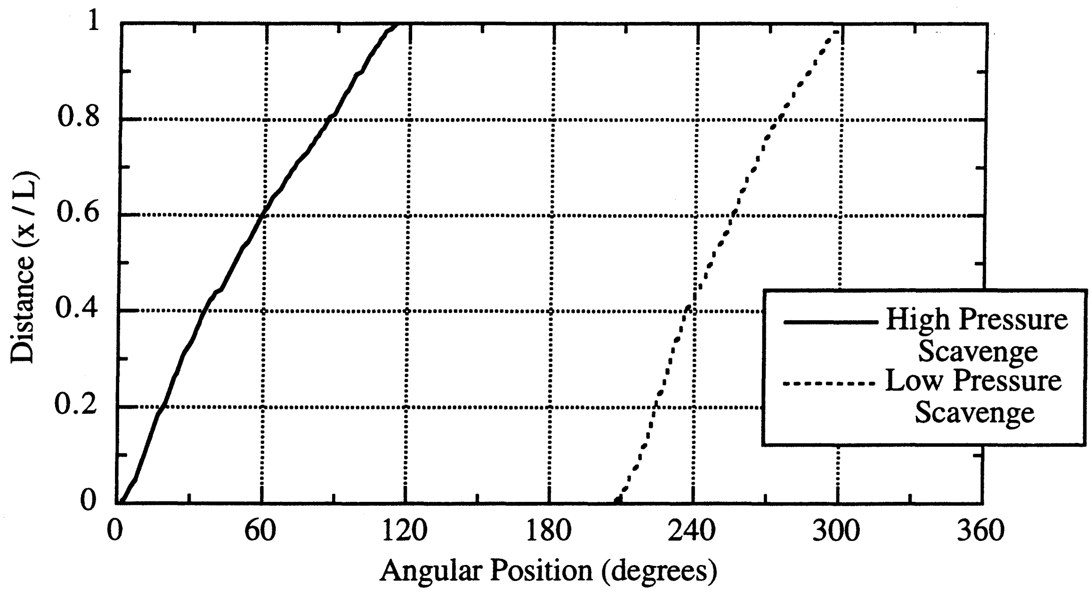


Figure 3.12 Contact Surface Position

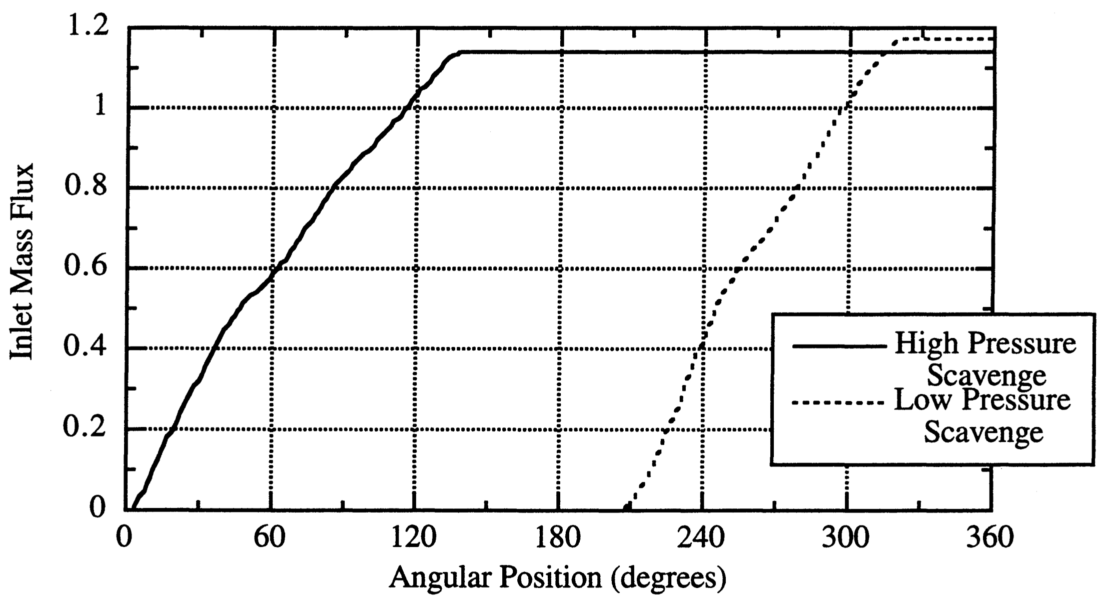


Figure 3.13 Mass Flux

Because the system is operating at steady-state, the high and low mass fluxes should be equal, but an error is noticable which can be attributed to several factors. First, the discrete step size will cause a slight variation due to interpolating which are compounded over the cycle. Also, the gradually opening and closing inlet ports do not precisely model the flow at the inlet which will result in slight errors. Finally, the actual inlet pressure is slightly less the the external stagnation pressure because the gas at the has a velocity which causes the inlet pressure to be a little lower.

3.10.5) Rotor Dimensions

If design specifications for a pressure exchanger require the rotor spin at 9000 RPM and ths device have a mass flow rate of 0.1 kg/s, the rotor will have a length of 15.8 cm and a frontal area of 30.4 cm^2. For an inner diameter of 5.0 cm, the outer diameter of the rotor will be 8.0 cm.

Chapter 4 Air-Cycle Model

4.0) Introduction

Air-cycle air conditioners use air as the refrigerant which eliminates the environmental damage that current CFC, HCFC and HFC air conditioners create. A basic air-cycle system can be developed with a pressure exchanger, an air-to-air heat exchanger and an air compressor. The system is modeled as an open Brayton refrigeration cycle, which consists of isentropic compression and expansion and constant pressure heat transfer.

4.1) Configurations

The pressure exchanger, heat exchanger and compressor can be integrated together in six different configurations to produce an air conditioning cycle. In three configurations, PC, SC1 and SC2, room air is compressed to a higher pressure and temperature to develop the temperature difference across the heat exchanger which allows energy transfer to the outside. With systems PE, SE1 and SE2, outside air is expanded below atmospheric pressure and temperature so that energy is drawn from the room.

4.1.1) System PC

In system PC (Figure 4.1), warm room air is drawn into the pressure exchanger and compressor. The two devices, operating in parallel, raise the pressure and temperature of the room air (1 & 3) above outside ambient conditions (2 & 4). The two air streams mix (5) and pass through the heat exchanger where energy is transferred to outside air. The air then returns to the pressure exchanger (6) where it expands back to atmospheric pressure and exits at a cooler temperature (9).

4.1.2) System SC1

In configuration SC1 (Figure 4.2), compression of room air (1) progresses in series; the air is compressed first by the compressor (2) and then by the pressure exchanger (4). Between the two compression stages, the air is cooled with a heat exchanger (3a). Any condensation that results is removed from the system (3b). After the pressure exchanger raises the air to its peak pressure, the air passes through a second heat exchanger (5). Finally, the air is expanded back to atmospheric pressure in the pressure exchanger (8). Condensation the develops in the second heat exchanger or in the expansion stage of the pressure exchanger is extracted before the air reenters the room.

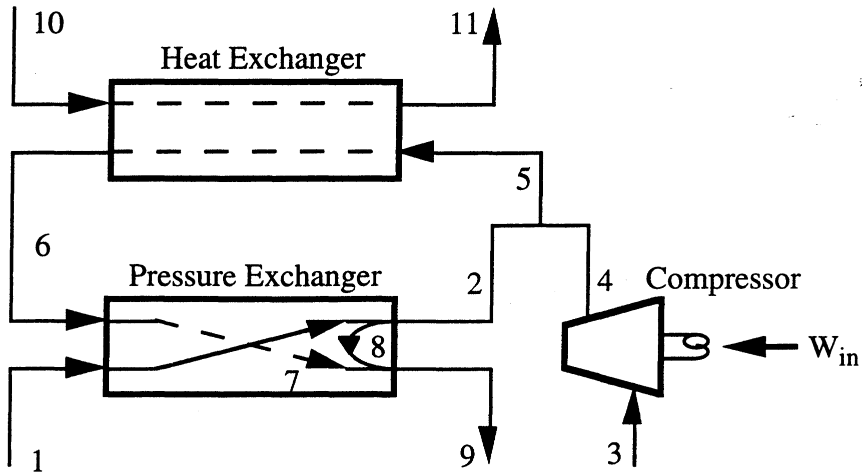


Figure 4.1 System PC

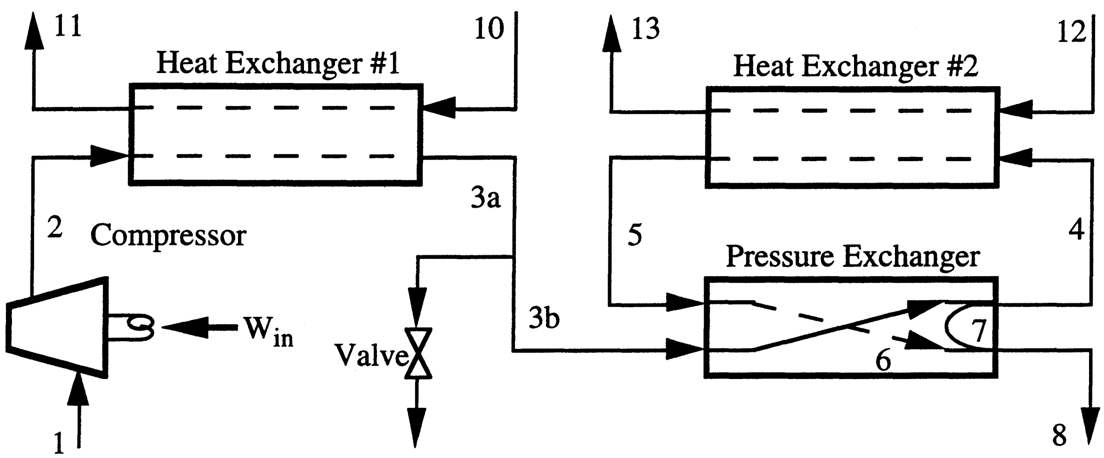


Figure 4.2 System SC1

4.1.3) System SC2

Systems SC1 and SC2 are alike in that the compressor and pressure exchanger compress the air in series, but the order has been reversed. In system SC2 (Figure 4.3), the room air (1) is first compressed by the pressure exchanger (2). Energy is removed by heat exchanger #1 (3a) and the air is compressed a second time by the compressor (4). A second heat exchanger transfers energy to the outside (5). The air finally expands back to atmospheric pressure and a cooler temperature (8). Condensation is removed before air enters the compressor (3b).

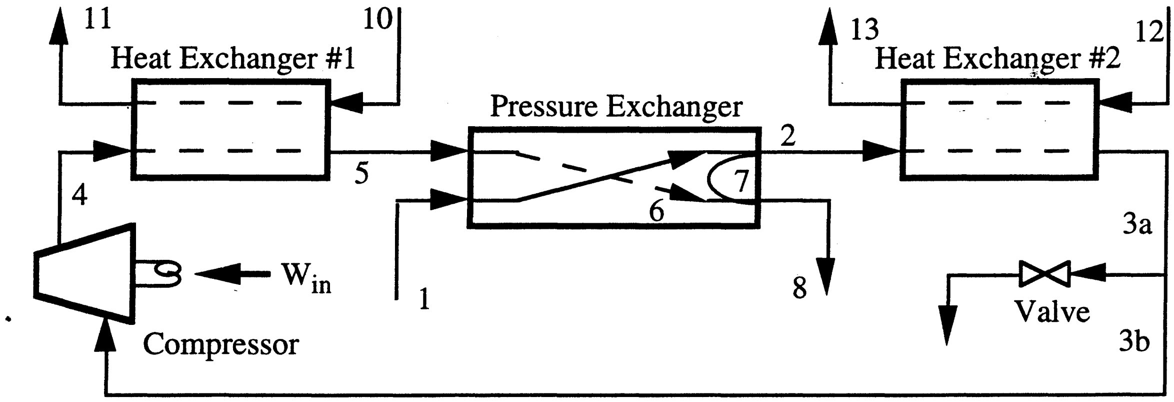


Figure 4.3 System SC2

4.1.4) System PE

In system PE (Figure 4.4), outside air (1) is initially expanded with the pressure exchanger to sub-atmospheric pressure and a low temperature (3). The cool air passes through a heat exchanger where it absorbs energy from the cooled space (4a). Moisture is removed from the system before compression begins (4b). The air stream then divides (5 & 8) so that the pressure exchanger and compressor can compress the parallel streams back to atmospheric pressure (6 & 9 respectively).

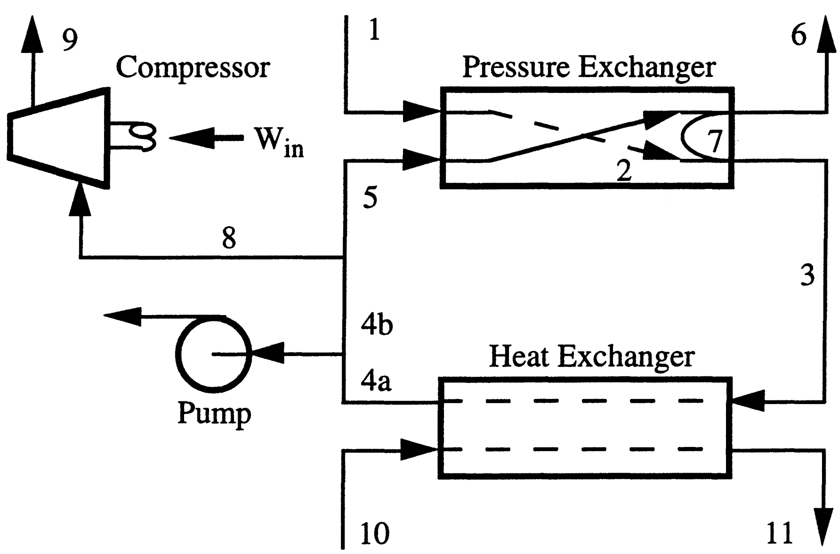


Figure 4.4 System PE

4.1.5) System SE1

The compressor and pressure exchanger are placed in series in this configuration (Figure 4.5). The first steps are the same as with the previous case: outside air (1) is expanded (3) and is passed through a heat exchanger (4a); condensation is removed (4b). Next the heated air is compressed by the compressor (5) and then the pressure exchanger (6). The depressurized systems only have one heat exchanger, unlike the pressurized, because the first compression stage raises the temperature above the point where additional energy can be transferred from the room.

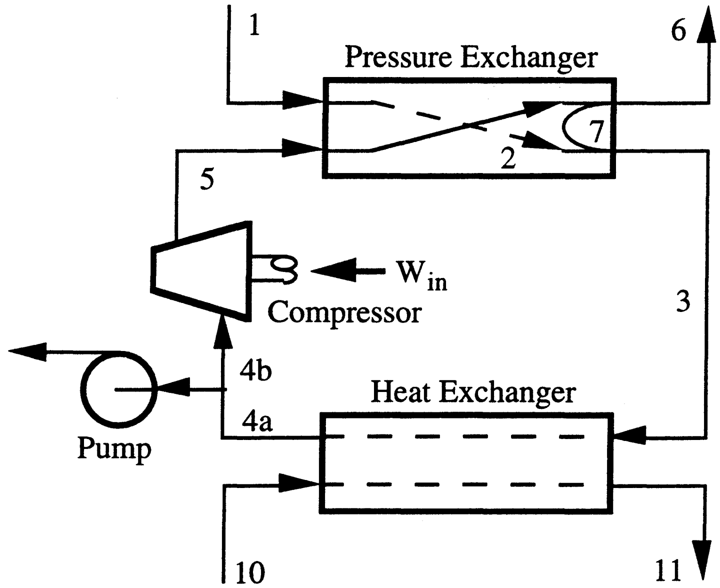


Figure 4.5 System SE1

4.1.6) System SE2

In this configuration (Figure 4.6), the compressor and pressure exchanger are again in series, but the compressor is located after the expansion and compression stages of the pressure exchanger. Outside air (1) is again expanded (3) and passed through a heat exchanger (4). This time, the pressure exchanger compresses (5) the air before the compressor (6).

4.2) Model Development

Development of the thermodynamic models for the three components and the system is important so that the performance of the different configurations can be simulated.

4.2.1) Pressure Exchanger

In a pressure exchanger, the energy released during expansion from one air stream compresses another. The pressure exchanger is modeled as two steady state processes, compression and expansion. The compression is the result of a normal shock. Because a normal shock is not isentropic, an isentropic compression efficiency for an ideal gas is calculated based on the pressure at the low pressure (LP) inlet and high pressure (HP) outlet ports:

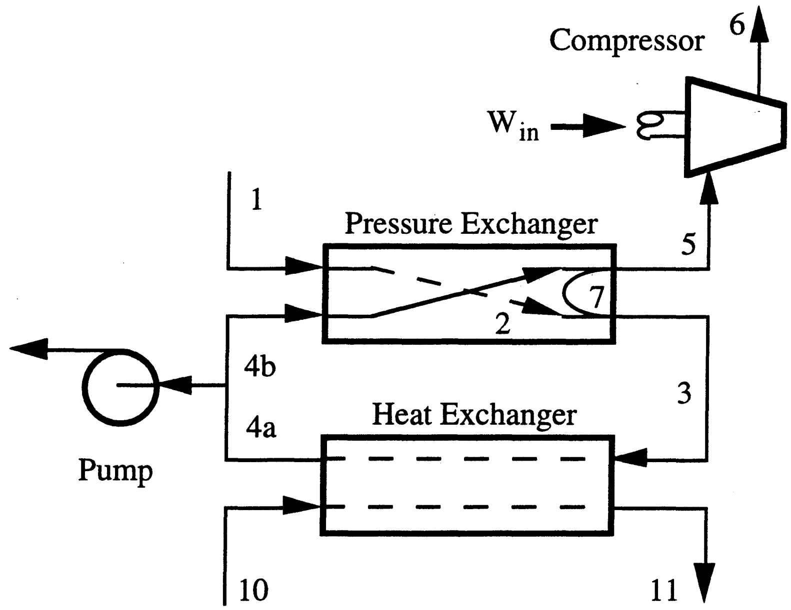


Figure 4.6 SE2

$$\frac{s_{HP} - s_{LP}}{R} = \frac{1}{\gamma - 1} ln\left[\frac{2\gamma M_x^2}{\gamma+1} - \frac{\gamma-1}{\gamma+1}\right] + \frac{\gamma}{\gamma-1} ln\left[\frac{2+(\gamma-1)M_x^2}{(\gamma+1)M_x^2}\right] \tag{4.1}$$

where:

$$\frac{P_{HP}}{P_{LP}} = \frac{2\gamma M_x^2 - (\gamma - 1)}{\gamma + 1} \tag{4.2}$$

Expansion waves are Mach waves which are reversible; therefore, the expansion process is modeled as isentropic. The energy released during expansion and the energy consumed during compression must be equal to satisfy the energy balance.

In addition to energy, the device must also conserve mass. Some simulations from the numerical model show that the high pressure outlet port will close because the flow reverses direction. When this occurs, the scavenge process is not complete and air is trapped that would have exited. Therefore, the trapped air remains in the system and exits through the low pressure port. In the thermodynamic model, all low pressure air is compressed to the outlet high pressure, but only a percentage of mass, based on the numerical model continues through the high pressure loop. The remaining mass expands back to the low pressure and mixes with the expanded high pressure air.

4.2.2) Air-to-Air Heat Exchanger

Typical air-to-air heat exchangers have an effectiveness between 70% and 90%, which limits the heat transfer rate. The model follows conservation of mass and energy. Pressure losses in air-to-air heat exchangers due to friction have been omitted from the model.

4.2.3) Air Compressor

The compressor is an integral part in the system because it adds energy to the system. With the parallel configurations (PC and PE), both high pressure ports and both low pressure ports are at the same pressure. Compression requires more energy per mass than the expansion releases. Therefore, the compressor adds mass to the high pressure side which will expand through the pressure exchanger satisfying the energy balance.

The mass flow rate is the constant throughout systems when the pressure exchanger and compressor are in series (SC1, SC2, SE1 and SE2). The energy from expanding the air from the high pressure to low pressure is used to compress the air, but because the expanding air has less energy, the compression process cannot achieve the high pressure. The compressor provides the additional boost the air needs to reach the peak pressure. The air compressor is modeled with an typical isentropic efficiency of 80%.

4.2.4) System

The pressure exchanger, air-to-air heat exchanger and air compressor can be combined to create different systems. Each system conserves mass and energy. It is also important to examine how each configuration performs. In order to compare the systems, nominal conditions inlet and outlet conditions for a unit with a one ton (3.52 kW) refrigeration capacity are established, which are:

- Ambient room temperature - 25°C
- Room relative humidity - 75%
- Ambient outside temperature - 35°C
- Room relative humidity - 75%
- System outlet temperature to room - 9°C
- Ambient pressure - 101.325 kPa

At these conditions, volumetric flow rates are set for each configuration allowing pressure to fluctuate so that mass and energy are conserved for the system.

4.3) Configuration Analysis

Each parameter is changed to a high and low value while all others remain constant to determine how the system performance changes. The high and low values are:

- Ambient room temperature - 20°C and 30°C
- Room relative humidity - 55% and 95%
- Ambient outside temperature - 30°C and 40°C
- Room relative humidity - 55% and 95%
- System outlet temperature to room - 9°C, 12°C and 15°C

The heat exchanger effectiveness will also affect the system performance. Effectiveness values of 70% and 90% are used.

Outside humidity does not affect system performance of the pressurized systems. Outside air, that passes through the cold side of heat exchanger, absorbs energy from the room air. Therefore, the temperature of the outside air exits the heat exchanger at a warmer temperature. Humidity becomes important with phase changes, either evaporation or condensation, because latent energy is consumed or released. For example, Table 4.1 shows no change in outlet temperature, percentage condensed moisture, and COP for configuration PC when the outside relative humidity is varied from 55% to 95%. The system performance at different outside relative humidities will not be discussed further.

$\phi_{outside}$	T_{out} (°C)	$\omega_{out} / \omega_{in}$	COP	P_{high} (kPa)	Q (kW)
55%	9.0	48.2%	3.45	185.0	3.52
75%	9.0	48.2%	3.45	185.0	3.52
95%	9.0	48.2%	3.45	185.0	3.52

Table 4.1 Effects of Outside Humidity on System PC Performance

In addition to checking specific points, plots for PC and SC2 configurations show trends over a broad range of room air (20°C to 35°C) and outside air (25°C and 45°C) temperatures. Six cases are run at room relative humidities of 55%, 75% and 95% and heat exchanger effectiveness of 70% and 90%. The sub atmosphereic pressure systems are not practical which will be shown.

4.3.1) System PC

The inlet room temperature (Table 4.2) affects the system performance. At low room temperatures, the air must be compressed to a higher pressure to achieve an energy balance. While this raises the temperature, the heat exchanger transfers less energy than at the high room temperature. This results in a lower COP for cool room air than for warm room air. Even though more energy is extracted from 30°C air than 20°C air (both have the same relative humidity), more moisture condenses at the cooler temperature. More water vapor is present at 30°C. Therefore, a smaller percentage of air at 30°C releases more latent energy.

T_{room} (°C)	T_{out} (°C)	$\omega_{out} / \omega_{in}$	COP	P_{high} (kPa)	Q (kW)
20	3.3	44.1%	3.04	190.7	3.369
25	9.0	48.2%	3.45	185.0	3.517
30	14.9	52.9%	3.88	180.4	3.672

Table 4.2 Effects of Room Temperature on System PC Performance

The humidity in the room (Table 4.3) affects the system performance due to condensation and the latent heat released. At lower inlet humidities, less moisture is present and more sensible energy can be removed by the heat exchanger which results in a lower temperature. On the other hand, more moisture condenses from air with a high moisture content. Therefore, less sensible energy is removed and the outlet temperature is warmer. Room humidity does not significantly affect COP because increases in compressor work are balanced by increases heat transfer.

ϕ_{room}	T_{out} (°C)	$\omega_{out} / \omega_{in}$	COP	P_{high} (kPa)	Q (kW)
55%	5.1	50.5%	3.41	181.5	3.363
75%	9.0	48.2%	3.45	185.0	3.517
95%	12.4	47.6%	3.50	188.7	3.676

Table 4.3 Effects of Room Humidity on System PC Performance

The outside inlet temperature (Table 4.4) has a small impact on the temperature. More moisture is removed at lower outside temperatures because more heat transfer occurs in the heat exchanger. The system COP increases as the amount of energy transferred increased.

$T_{outside}$ (°C)	T_{out} (°C)	$\omega_{out} / \omega_{in}$	COP	P_{high} (kPa)	Q (kW)
30	8.0	44.9%	3.86	180.1	3.641
35	9.0	48.2%	3.45	185.0	3.517
40	10.0	51.5%	3.07	190.6	3.395

Table 4.4 Effects of Outside Temperature on System PC Performance

The heat exchanger effectiveness (Table 4.5) is significant because it affects the rate of heat transfer. The hot side temperature must be increased so that the system maintains the nominal refrigeration capacity. This is accomplished by raising the high pressure which causes a decrease in COP. The mass flow rate decreases with effectiveness because the high pressure gas entering the pressure exchanger has more energy.

ε	V_{in} (m^3/s)	$\omega_{out} / \omega_{in}$	COP	P_{high} (kPa)
70%	0.05449	48.2%	2.10	224.9
90%	0.06185	48.2%	3.45	185.0

Table 4.5 Effects of Heat Exchanger Effectiveness on System PC Performance

As the nominal outlet temperature (Table 4.6) increases, the volumetric (mass) flow rate through the pressure exchanger increases while the volumetric (mass) flow rate through the compressor remains steady. As a result, COP increases as the nominal outlet temperature rises. A trade-off exists between improved COP and reduced condensation because the dew point temperature is higher.

T_{out} (°C)	V_{PXchr} (m^3/s)	V_{Comp} (m^3/s)	$\omega_{out} / \omega_{in}$	COP	P_{high} (kPa)
9	0.05292	0.00894	48.2%	3.45	185.0
12	0.06925	0.00926	59.0%	4.11	167.7
15	0.09838	0.00976	72.0%	5.03	150.7

Table 4.6 Effects of Heat Exchanger Effectiveness on System PC Performance

The trends discussed above are shown in plots in Appendix C.1. The ϕ_6 = 100% line and the area to the lower right delineates where condensation occurs in the heat exchanger. Otherwise, moisture only condenses during the expansion stage of the pressure exchanger.

4.3.2) System SC1

This configuration is not realistic because the pressure difference between the high and low pressure inlets is smaller than the other configurations which causes the generation of weaker normal shocks. As a result of the shock waves moving at a slower velocity, the high pressure scavenge is unable to finish before the high pressure outlet port closes. Therefore, a portion of the entering low pressure air leaves through the low pressure outlet at a warmer temperature, not having passed through the heat exchanger. The system tries to compensate by increasing the ratio of pressure rise across the compressor to pressure rise across the pressure exchanger which raises the temperature so that more energy can be removed through the first heat exchanger. Unforturnately, most of the compression must take place in the compressor to achieve an equilibrium which drastically lowers the COP.

4.3.3) System SC2

As the room air inlet temperature (Table 4.7) increases, the temperature difference across the heat exchanger increases which enables more energy to leave the system, which translates into better COP. For a 5°C change in room air temperature, the outlet air changes by about 5.5°C and percent moisture condensation changes by 2%. The lower heat transfer rate is offset because cool air has less energy than warm air. Therefore, a variation of 5°C room temperature almost directly translates to the change in outlet temperature.

T_{room} (°C)	T_{out} (°C)	$\omega_{out} / \omega_{in}$	COP	P_{int} (kPa)	P_{high} (kPa)	Q_1 (kW)	Q_2 (kW)
20	3.9	45.7%	2.80	163.0	179.9	2.16	1.05
25	9.0	47.8%	3.13	160.4	177.3	2.45	1.07
30	14.5	51.0%	3.48	157.9	174.8	2.71	1.09

Table 4.7 Effects of Room Temperature on System SC2 Performance

The relative humidity in the room (Table 4.8) is important to the performance of the SC2 configuration. The moisture content in the air affects the ratio between sensible and latent energy removed. At higher relative humidities, more latent energy is released to condense the moisture in the air; therefore, the outlet temperature is warmer. One benefit of more moisture is a slight improvement in COP.

ϕ_{room}	T_{out} (°C)	$\omega_{out} / \omega_{in}$	COP	P_{int} (kPa)	P_{high} (kPa)	Q_1 (kW)	Q_2 (kW)
55%	4.3	47.4%	3.02	160.4	177.9	2.45	1.10
75%	9.0	47.8%	3.13	160.4	177.3	2.45	1.07
95%	13.1	49.3%	3.25	160.3	176.8	2.45	1.04

Table 4.8 Effects of Room Humidity on System SC2 Performance

The outside temperature (Table 4.9) has a greatest impact on the amount of energy removed from the system. With low outside temperatures, the temperature difference (driving potential) across the heat exchanger is greater and more sensible and latent energy is transferred. The system COP improves due to the higher heat transfer.

$T_{outside}$ (°C)	T_{out} (°C)	$\omega_{out} / \omega_{in}$	COP	P_{int} (kPa)	P_{high} (kPa)	Q_1 (kW)	Q_2 (kW)
30	7.4	42.8%	3.47	158.0	174.8	2.71	1.09
35	9.0	47.8%	3.13	160.4	177.3	2.45	1.07
40	10.5	52.9%	2.81	162.8	179.8	2.18	1.05

Table 4.9 Effects of Outside Temperature on System SC2 Performance

As with the previous configurations, the heat exchanger effectiveness (Table 4.10) plays a key role in the performance of the system. For low ε, the peak system pressure and temperature must be raised so that the driving potential across the heat exchanger will compensate for the lower transfer rate. The result is a lower COP.

ε	V_{in} (m^3/s)	COP	P_{int} (kPa)	P_{high} (kPa)
70%	0.06700	2.74	169.4	189.4
90%	0.06933	3.13	160.4	177.3

Table 4.10 Effects of Heat Exchanger Effectiveness on System SC2 Performance

Higher pressures are necessary to achieve lower nominal outlet temperatures (Table 4.11). The mass flow rate also decreases at lower outlet temperature because more energy is removed from less air at the same cooling capacity. Between 9°C and 12°C, the work needed to compress the air over the pressure differential balances which results in COP remaining constant. As outlet temperature approaches 15°C, the increase in volumetric flow rate increases faster than the decrease in pressure rise across the compressor. As a result, the compressor must do more work, and the COP decreases. Less moisture condenses at higher outlet temperature because the dew point is lower.

T_{out} (°C)	V_{in} (m^3/s)	$\omega_{out} / \omega_{in}$	COP	P_{int} (kPa)	P_{high} (kPa)	Q_1 (kW)	Q_2 (kW)
9	0.06933	47.8%	3.13	160.4	177.3	2.45	1.07
12	0.08807	58.5%	3.11	149.5	162.1	2.43	1.09
15	0.1231	71.3%	2.87	137.9	147.1	2.35	1.17

Table 4.11 Effects of Heat Exchanger Effectiveness on System SC2 Performance

The plots showing the performance of configuration SC2 are in Appendix C.2. The beginning of condensation in heat exchangers #1 and #2 are marked. Moisture continues to condenses as the room air becomes warmer and/or the outside air becomes cooler. A discontinuity exists in the lines representing constant outlet temperature lines and constant humidity ratio when condensation begins in heat exchanger #1 because that condensate is removed before compression in the compressor.

4.3.4) Systems PE, SE1 and SE2

The three configurations which expand outside air below ambient pressure are not practical systems. To achieve an outlet air temperature (Table 4.12) of 9°C, the low temperatures in the systems are well below freezing which leads to frosting.

System	T_{out} (°C)	V_{in} (m^3/kg)	$\omega_{out} / \omega_{in}$	COP	T_{low} (°C)	P_{low} (kPa)
PE	9	0.08471	71.3%	0.717	-13.7	25.3
SE1				0.842	-14.6	24.7
SE2				0.600	-14.4	24.8
PE	15	0.1454	47.8%	1.36	2.5	42.5
SE1				1.46	2.2	42.2
SE2				0.641	2.6	42.3

Table 4.12 Effects of Outlet Temperature on Systems PE, SE1 and SE2 Performance

Most frosting can be eliminated by setting the nominal cooling temperature at 15°C, which is warm for outlet air. The system adjusts to the warmer outlet condition by expanding the outside air to 42 kPa, instead of 25 kPa. Therefore, the low temperature remains above freezing. In order to sustain the same cooling load, the mass flow rate through the system must increase 70%.

Other factors make expanding outside impractical. The internal ducts must be larger to prevent the flow from choking because of the low density of sub-atmospheric air. It is also difficult to remove water vapor from air in a vacuum. Also, outside humidity adds great variability to system performance which is undesirable. In Table 4.13, the performance of configuration PE is shown with different humidity levels. Because moisture in the outside air condenses during expansion in the pressure exchanger, more latent energy and less sensible energy are extracted from the air. Therefore, the inlet temperature for the cold side of the heat exchanger can increase 10°C with a 40% rise in relative humidity. The systems where outside air is expanded will not be pursued further.

$\phi_{outside}$	T_{out} (°C)	$\omega_{out} / \omega_{in}$	COP	T_{low} (°C)	P_{low} (kPa)
55%	13.4	64.4%	1.63	-2.0	42.7
75%	15.0	71.3%	1.36	2.5	42.5
95%	16.7	79.4%	1.05	7.6	42.5

Table 4.13 Effects of Outside Humidity on System PE Performance

4.4) Configuration Comparison

With the performance trends, configurations PC and SC2 can be compared to determine which system operates better.

4.4.1) Nominal Condition

At nominal conditions, both systems are operating with the same inlet and outlet conditions (Table 4.14). The mass flow rate for PC is less than SC2 while the COP for PC is 10% better than SC2.

System	V_{in} (m^3/s)	COP
PC	0.06185	3.45
SC2	0.06933	3.13

Table 4.14 Performance Comparison at Nominal Conditions

4.4.2) Room Temperature

Both systems have outlet temperature within 1°C and moisture condensation percentage within 2% of each other for warm and cool room air inlet temperatures (Table 4.15). The COP of SC2 decreases from 92% of PC to 90% when operating with a higher inlet room temperature.

System	T_{room} (°C)	T_{out} (°C)	$\omega_{out} / \omega_{in}$	COP
PC	20	3.3	44.1%	3.04
SC2		3.9	45.7%	2.80
PC	30	14.9	52.9%	3.88
SC2		14.5	51.0%	3.48

Table 4.15 Performance Comparison of Different Room Temperatures

4.4.3) Room Inlet Relative Humidity

While room humidity (Table 4.16) affects the performance of each configuration, little distinction exists between the systems. At the two humidity levels, 55% and 95%, the outlet air temperature varies by less than 1°C and the percent condensation by about 3%. While PC has the highest COP, SC2 narrows the gap by 5% at higher relative humidities.

System	ϕ_{room}	T_{out} (°C)	$\omega_{out} / \omega_{in}$	COP
PC	55%	5.1	50.5%	3.41
SC2		4.3	47.4%	3.02
PC	95%	12.4	47.6%	3.50
SC2		13.1	49.3%	3.25

Table 4.16 Performance Comparison at Different Room Relative Humidities

4.4.4) Outside Inlet Temperature

The outlet temperature and percent moisture removed vary by about 1°C and 2%, respectively, between each configuration at a set outlet temperature (Table 4.17). The COP for configuration PC is 11% and 9% higher than configeration SC2 at low and high outside temperatature, respectively.

System	$T_{outside}$ (°C)	T_{out} (°C)	$\omega_{out} / \omega_{in}$	COP
PC	30	8.0	44.8%	3.86
SC2		7.4	42.9%	3.47
PC	40	10.0	51.5%	3.07
SC2		10.5	52.9%	2.81

Table 4.17 Performance Comparison at Different Outside Temperatures

4.4.5) Heat Exchanger Effectiveness

Heat exchanger effectiveness (Table 4.18) affects the COP for all the configurations. At 90% effectiveness, system PC preforms 10% better than SC2. But when less efficient heat exchangers are used in the systems, SC2 is better 30% better than PC.

System	ε	V_{in} (m^3/s)	COP
PC	70%	0.05449	2.10
SC2		0.06700	2.74
PC	90%	0.06185	3.45
SC2		0.06933	3.13

Table 4.18 Performance Comparison at Different Heat Exchanger Effectiveness

4.4.6) Outlet Air Temperature

The nominal outlet temperature (Table 4.19) may be varied from what was used as the example. Setting the outlet temperature lower than 9°C will result in frosting and should be avoided. As the outlet temperature increases, the COP for configuration PC increases from 10% better than SC2 at 9°C to 75% better at 15°C.

System	T_{out} (°C)	V_{in} (m^3/s)	COP
PC	9	0.06185	3.45
SC2		0.06933	3.13
PC	12	0.07850	4.11
SC2		0.08807	3.11
PC	15	0.1081	5.03
SC2		0.1231	2.87

Table 4.19 Performance Comparison at Different Nominal Outlet Temperatures

Comparing the performance of PC and SC2, neither configuration appears superior at lowering the temperature and removing moisture. The COP for PC is slightly higher than SC2, but this is dependent on the operation conditions and heat exchanger effectiveness.

Configuration PC has several non-performance advantages. The system has one heat exchanger before the air expands in the pressure exchanger and exits the system. Thererfore, only one condensation trap is needed at the exit to remove the condensate.

Chapter 5 Summary

5.0 Introduction

In the previous chapter, the pressure exchanger is modeled as a general compression and expansion device. The steady-state operating conditions determined by the air-cycle model enable the numerical model predict the port locations and rotor dimensions. This leads to future work for developing an operating pressure exchanger-based air-cycle air conditioner.

5.1 Pressure Exchanger Dimensions

From the numerical model, the port locations are determined for nominal conditions (Table 5.1). For system PC, the high pressure scavenge occurs over 138° of one rotation (360°) followed by all ports remaining closed for 41°. The low pressure scavenge inflow takes place for 150° of the cycle after which the ports stay closed for 31°. Then, the next cycle begins. Similarly for configuration SC2, the high pressure scavenge proceeds over 133° and the low pressure scavenge over 171° with no ports closed for 14° and 42°.

System	High Pressure Ports		Low Pressure Ports	
	Inlet	Outlet	Inlet	Outlet
PC	0.0° - 138.1°	23.4° - 118.4°	214.5° - 329.2°	178.8° - 307.0°
SC2	0.0° - 132.5°	25.9° - 106.9°	190.1° - 318.2°	146.9° - 292.3°

Table 5.1 Pressure Exchanger Port Locations

The length of the rotor is dependent on the rotor's rotational speed, as shown in Table 5.2. At slower angular speeds, the rotor must be lengthened so that the normal shocks, expansion waves and contact surfaces occur at the proper location in the cycle. Lengthening the rotor allows for more rotation so that the waves match up with the port locations. For configuration PC, the rotor length is 23.7 cm at 6000 RPM and shortens to 9.5 cm at 15,000 RPM.

System	Length (cm)				Frontal Area
	6000 RPM	9000 RPM	12000 RPM	15000 RPM	(cm^2)
PC	23.7	15.8	11.9	9.5	19.0
SC2	27.7	18.5	13.8	11.1	21.0

Table 5.2 Rotor Dimensions

The frontal area of the rotor cells, which regulates the mass flow into and out of the system, is independent of rotational speed because at higher speeds the ports are open for a shorter time allowing less mass to enter a cell. If the inner diameter of the rotor for configuration PC is 5.0 cm, then the outer diameter is 7.0 cm.

5.2 Dimensional Variations

It is important to examine how the pressure exchanger will change when the operating conditions vary. For system PC (Table 5.3), the peak pressure experiences a low (171 kPa) when the room inlet air is 30°C and 55% relative humidity and the outside inlet air is 30°C based on the parameters tested in Chapter 4. The normal shocks become weaker with the smaller pressure gradient and they propagate through the air at a slower velocity. Therefore, the inlet and outlet high pressure ports must be open for a greater portion of the cycle so that the contact surface can traverse the cell. At the high peak pressure (200 kPa) where the room inlet temperature is 20°C and relative humidity is 95% and the outside inlet temperature is 40°C, the normal shocks are much stronger and the high pressure scavenge process proceeds at a faster velocity.

High Pressure (kPa)	High Pressure Inlet	High Pressure Outlet	Low Pressure Inlet	Low Pressure Outlet
170.9	0.0° - 159.3°	23.1° - 138.7°	208.0° - 329.6°	173.9° - 307.7°
185.0	0.0° - 138.1°	23.4° - 118.4°	214.5° - 329.2°	178.8° - 307.0°
199.7	0.0° - 130.2°	24.3° - 108.5°	210.6° - 323.0°	172.3° - 301.3°

Table 5.3 Pressure Exchanger Port Locations for System PC

The ideal dimensions of the rotor also change with the different inlet conditions (Table 5.4). When the angular speed of the rotor is constant, the length of the rotor must be shorter when the peak pressure is 171 kPa instead of 200 kPa because the high pressure scavenge proceeds more slowly which requires the shorter rotor so that the contact surface reaches the outlet port before is closes. Also, the frontal area of the rotor decreases with as the peak pressure rises because the pressure gradient is larger cause higher fluid velocities.

High Pressure (kPa)	Length (cm) 6000 RPM	9000 RPM	12000 RPM	15000 RPM	Frontal Area (cm^2)
170.9	23.4	15.6	11.7	9.4	19.2
185.0	23.7	15.8	11.9	9.5	19.0
199.7	24.6	16.4	12.3	9.9	18.3

Table 5.4 Rotor Dimensions for System PC

5.3 Future Work

Several steps should be taken next:

- The unsteady flow model should be experimentally validated. While the method of characteristics is an accepted technique, several assumptions are made which introduce errors, some of which might be significant.
- The homentropic assumption should be relaxed because the high pressure and low pressure inlets are at different entropies which violates the underlining principle of constant entropy throughout homentropic flow.
- The air-cycle model shows that most moisture condenses in the pressure exchanger during the expansion phase. The effect of condensation should be added to the unsteady flow model to determine what effect condensation has on the performance of the pressure exchanger.
- The pressure exchanger does not continually operate under the same inlet conditions. The changes in pressure exchanger performance should be determined and if the significance solutions should be explored.
- A detailed model of the air-cycle system should be constructed in which the components are modeled more closely to real world and other components, like a trap, are included.
- Integrate pressure exchanger and air-cycle models into one?
- Any others?

5.4 Conclusions

A numerical model was devloped using the method of characteristics to simulate unsteady flow through one cell of the pressure exchanger. The flow properties, port locations, mass flow rates and the rotor dimensions are calculated by the model. Several simplifying assumptions were made including one-dimensional, homentropic flow and ideal gas which eliminates condensation.

Six simple air-cycle air conditioning model were developed consisting of a pressure exchanger, an air compressor and one or two air-to-air heat exchanger of which configurations PC and SC1 exhibited better performance.

To improve the accuracy of both models, the homentropic flow and no condensation assumptions should be relaxed in the numerical pressure exchanger model and detailed component integrated into the air-cycle model.

References

1. Chase, Victor. "Here's a Chilling Thought: How to Keep Cool Without CFC's." Research and Development. Vol. 37, No. 11, Oct. 1995. pp. 39 - 40.

2. Jain, S., Dhar, P. L., and Kaushik, S. C. "Evaluation of Solid-Desiccant-based Evaporative Cooling Cycles for Typical Hot and Humid Climates." International Journal of Refrigeration. Vol. 18, No. 5, 1995. pp. 287 - 296.

3. Engelking, S. and Kruse, H. "Development of Air Technology for Transport Refrigeration." 1996 International Refrigeration Conferance at Purdue, West Lafayette, Indiana. July 23 - 26, 1996. pp. 349 - 356.

4. Kentfield, J. A. C. "The Pressure Exchanger: An Introduction Including a Review of the Work of Power Jets (R & D) Ltd." Proceedings of the 1985 ONR/NAVAIR Wave Rotor Research and Technology Workshop, Naval Postgraduate School, Monterey, California. pp. 10 - 49.

5. Weber, H. E. "Shock-Expansion Wave Engines - New Directions for Power Production." Proceedings of the 1985 ONR/NAVAIR Wave Rotor Research and Technology Workshop, Naval Postgraduate School, Monterey, California. pp. 195 - 214.

6. Berchtold, M. "Supercharging with Comprex." Proceedings of the 1985 ONR/NAVAIR Wave Rotor Research and Technology Workshop, Naval Postgraduate School, Monterey, California. pp. 51 - 74.

7. Zucrow, M. J. and Hoffman, J. D. Gas Dynamics, Vol. 1. John Wiley & Sons, New York, 1976.

8. Saad, Michel A., Compressible Fluid Flow, 2nd Ed. Prentice Hall, Englewood Hills, New Jersey. 1993. pp. 149 - 150.

9. Addy, A. L., Morris, M. J., and Dutton, J. C. "An Investigation of Compressible Flow Characteristics of Butterfly Valves." Journal of Fluids Engineering. Vol. 107, Dec. 1985. pp. 512 - 517.

Appendix A Numerical Model

```
      PROGRAM MAIN
c
c  Modified from program COMPF.BAS,  HHK 02/23/1994
c  This program is the final version except for C-D nozzle inflow - 08/22/96
c
       double precision gamma, P_end(4), P, U, X, Y_l, Y_h, mass, eps,
     &      dt, rate, Aet(2)
       integer c, i, j, no, nx, nj, Pt, part(2), counter, next, loc(4),
     &      cp(4)
       common/aa/gamma, dt, eps, nx, j
       common/bb/Y_l(500), Y_h(500), mass(500), Pt(4,2)
       common/cc/P(11,500), U(11,500), X(11)
c
c  INPUT VARIABLES
c
c  gamma - specific heat ratio
c  P_end(1) - high pressure inflow
c  P_end(2) - high pressure discharge
c  P_end(3) - low pressure discharge
c  P_end(4) - low pressure inflow
c
c  dt - time step size
c  nj - number of time steps
c  no - number of initial left end outflow steps
c  nx - number of stations in the tube
c
c  INTERMEDIATE VARIABLES
c
c  j - number of the current step
c  Aet - area ratio for a partially open port
c  part - time step since port opening
c  conv - convergence counter (0 if converged; 1 if not)
c
c  loc - 1:  left end high pressure port is open
c        2:  left and right end high pressure ports are open
c        3:  right end high pressure port is open
c        4:  all ports are closed
c        5:  right end low pressure port is open
c        6:  left and right end low pressure ports are open
c        7:  left end low pressure port is open
c        8:  all ports are closed
c
c  SOLUTION VARIABLES
c
c  P(nx, nj) - pressure at station 'nx' and time 'nj'
c  U(nx, nj) - velocity at station 'nx' and time 'nj'
c  X(nx) - location of station 'nx' from left end
c  Y_h(nj) - location of interphase line in the tube at high pressure
```

```
c  Y_l(nj) - location of interphase line in the tube at low pressure
c  mass(nj) - mass flow rate exiting the tube
c
c  METHOD OF CHARACTERISTICS
c
c  P1, U1, X1, A1 - right-running characteristic intersecting initial value line
c  P2, U2, X2, A2 - left-running characteristic intersecting initial value line
c  P4, U4, A4 - solution point (where left- and right-running char cross)
c  P5, U5, X5 - previous solution points to station left of current station
c  P6, U6, X6 - previous solution points at current station
c  P7, U7, X7 - previous solution points to station right of current station
c
c
      call INPUT (P_end, rate, nj, loc, cp)
c
      do 100 counter = 1, 50
      write(*,*) counter
       write(*,200) Pt(1,1), Pt(1,2)
       write(*,200) Pt(2,1), Pt(2,2)
       write(*,200) Pt(3,1), Pt(3,2)
       write(*,200) Pt(4,1), Pt(4,2)
c       pause
c
      Aet(1) = 0.
      Aet(2) = 0.
      Y_l(1) = 0.
      Y_h(1) = 0.
      mass(1) = 0.
      next = 0
      part(1) = 1
      part(2) = 100
      loc = 0
c
      do 101 j = 1, nj
c
c  Subroutine called which determines which ports are opened or closed
c
       call PORT (P_end, part, loc, cp, nj, next, rate, Aet)
c
c  Subroutines called which calculate properties at left end
c
       call LEFT (P_end, Aet(1), loc(4))
c
c  Subroutine called which calculates interior properties
c
       do 102 c = 2, nx-1
        call INTERIOR (c)
102     continue
c
c  Subroutine called which calculates properties at right end
c
       call RIGHT (P_end, Aet(2), loc(3))
c
c  Subroutine called which calculates the mass flow in the low pressure port
```

```
c
        call MASSFLOW (mass)
c
c  Subroutine called which calculates the location of the interphase line
c
        call DISTANCE (Y_h, U)
c
      if (loc(4) .gt. 0) then
        call DISTANCE (Y_l, U)
c
      else
        Y_l(j+1) = 0.
c
      end if
c
      if (next .eq. 1) then
        goto 100
      end if
c
c     if (loc(2) .ne. 0) then
c     write(*,*) j+1, dt * j, loc
c      write(*,*) 1, P(1,j+1), U(1,j+1)
c      write(*,*) 2, P(2,j+1), U(2,j+1)
c      write(*,*) 3, P(3,j+1), U(3,j+1)
c      write(*,*) 4, P(4,j+1), U(4,j+1)
c      write(*,*) 5, P(5,j+1), U(5,j+1)
c      write(*,*) 6, P(6,j+1), U(6,j+1)
c     write(*,*) mass(j+1), Y_l(j+1), Y_h(j+1)
c     pause
c     end if
101   continue
100   continue
c
      write(*,*) 'Solution did not converge'
c     write(*,*) j
c     call OUTPUT
c
200   format (i3,3x,i3,3x)
      stop
      end
c
c**********************************************************************
c
      SUBROUTINE INPUT (P_end, rate, nj, loc, cp)
c
      double precision gamma, P_end(4), P, U, X, Ui, Pi, eps, dt, rate
      integer j, no, nj, nx, loc, cp(4)
      common/aa/gamma, dt, eps, nx, j
      common/cc/P(11,500), U(11,500), X(11)
c
      open(unit = 20, file = 'data', status = 'old')
c
      gamma = 1.4
      nx = 6
```

```
      nj = 490
c
      P_end(1) = 2.
      P_end(2) = 2.
c      P_end(1) = .95 * P_end(2)
      P_end(3) = 1.
      P_end(4) = 1.
c      P_end(4) = .95 * P_end(3)
c
      rate = .2
      loc = 8
      dt = .075
      eps = .00001
      read(20,*) nx, rate, dt, eps
      read(20,*) P_end(1), P_end(2), P_end(3), P_end(4)
      read(20,*) cp(1), cp(2), cp(3), cp(4)
c
      Ui = 0.
c      Pi = (P_end(1) + P_end(3)) / 2.
      Pi = (P_end(1) + 1. * P_end(3)) / 2.
c
c      write(*,*) 'Enter the specific heat ratio, gamma:'
c        read(*,*) gamma
c      write(*,*) 'Enter discharge air, left, PA (Pa):'
c        read(*,*) PA
c      write(*,*) 'Enter high press. gas in, right end, PR (Pa):'
c        read(*,*) PR
c
c      write(*,*) 'Number of stations (incl. x=0 and x=1):'
c        read(*,*) nx
      write(*,*) 'Time step, delta t (sec) <', 1 / ((nx - 1.) * 2.)
     &     , ' for numerical stability'
c      write(*,*) 'Enter delta t (sec):'
c        read(*,*) dt
c      write(*,*) 'Enter # of time steps (<500):'
c        read(*,*) nj
c
c  An initial pressure and velocity are given to start the iterative
c    process
c
       do 100 i = 1, nx
         U(i,1) = Ui
         P(i,1) = Pi
         X(i) = (i - 1.) / (nx - 1.)
100    continue
c
      return
      end
c
c*******************************************************************
c
      SUBROUTINE OUTPUT
c
      double precision gamma, Y_l, Y_h, mass, P, U, X, dt, eps
```

```
      integer j, nx, i, m, n, Pt
      common/aa/gamma, dt, eps, nx, j
      common/bb/Y_l(500), Y_h(500), mass(500), Pt(4,2)
      common/cc/P(11,500), U(11,500), X(11)
c
      write(*,*) 'INTO OUTPUT'
c
c The output files are opened
c
      open(unit = 1, file = 'station1', status = 'unknown')
      open(unit = 2, file = 'station2', status = 'unknown')
      open(unit = 3, file = 'station3', status = 'unknown')
      open(unit = 4, file = 'station4', status = 'unknown')
      open(unit = 5, file = 'station5', status = 'unknown')
      open(unit = 6, file = 'station6', status = 'unknown')
      open(unit = 7, file = 'station7', status = 'unknown')
      open(unit = 8, file = 'station8', status = 'unknown')
      open(unit = 9, file = 'station9', status = 'unknown')
      open(unit = 10, file = 'station10', status = 'unknown')
      open(unit = 11, file = 'station11', status = 'unknown')
      open(unit = 12, file = 'distance', status = 'unknown')
      open(unit = 13, file = 'mass', status = 'unknown')
      open(unit = 14, file = 'ports', status = 'unknown')
c
c  The pressure and velocity are recorded at each station
c
     do 111 k = 1, nx
       write(k,*) ' Time    ', 'Position    ', 'Pressure       ',
    &   'Velocity'
      do 110 i = 1, j
        write(k,150) (i - 1) * dt, 1. * (i - 1) / (Pt(1,1) - 1.) *
    &      360., P(k,i), U(k,i)
110    continue
111   continue
c
c  The high and low pressure interphase lines and the mass flux are
c   recorded
c
       write(12,*) ' Time    ', 'Position    ',  'Y_l          ',
    &   'Y_h'
       write(13,*) ' Time    ', 'Position     ', 'Mass'
     do 120 m = 1, j
      write(12,150) (m - 1) * dt, 1. * (m - 1) / (Pt(1,1) - 1.) *
    &   360., Y_l(m), Y_h(m)
      write(13,151) (m - 1) * dt, 1. * (m - 1) / (Pt(1,1) - 1.) *
    &   360., mass(m)
120  continue
c
c  The locations for the opening and closing of ports is saved
c
     write(14,*) 'Locations for port opening and closing'
     do 130 n = 1, 4
      write(14,152) 1. * (Pt(n,1) - 1.) / (Pt(1,1) - 1.) * 360., 1. *
    &    (Pt(n,2) - 1.) / (Pt(1,1) - 1.) * 360.
```

```
130  continue
c
150   format (f7.3, 3x, f7.3, 3x, f11.7, 3x, f11.7)
151   format (f7.3, 3x, f7.3, 3x, f11.7)
152   format (f6.2,3x,f6.2)
c
      stop
      end
c
c*******************************************************************
c
      SUBROUTINE LEFT (P_end, Aet, loc)
c
c  This subroutine prepares the initial values for MOC at the left end
c
      double precision gamma, dt, eps, P, U, X, X6, X7, P6, P7, U6, U7,
     &       X2, P4, U4, P_end(4), Aet
      integer loc, nx, j
c
      common/aa/gamma, dt, eps, nx, j
      common/cc/P(11,500), U(11,500), X(11)
      common/dd/X6, X7, P6, P7, U6, U7, X2(2), P4(2), U4(2)
c
c  Initial value line established for MOC
c
      X6 = X(1)
      X7 = X(2)
      P6 = P(1,j)
      P7 = P(2,j)
      U6 = U(1,j)
      U7 = U(2,j)
c
c  Solution points are given initial guess
c
      X2(1) = X7
      X4 = X6
      P4(1) = P6
      U4(1) = U6
c
c  Proper boundary condition is called
c
      if (Aet .eq. 0.) then
        U4(1) = 0.
        call CLOSED_L
c
      else if (loc .eq. 0) then
        if (U(1,j) .lt. 0.) then
          U4(1) = -U4(1)
          U6 = -U6
          U7 = -U7
          X6 = 1.
          X7 = .8
          call OUTFLOW (P_end(1), Aet)
          U4(2) = -U4(2)
```

```
         else
          call INFLOW (P_end(1), Aet)
         end if
c
       else if (loc .gt. 0) then
        if (U(1,j) .lt. 0.) then
         U4(1) = -U4(1)
         U6 = -U6
         U7 = -U7
         X6 = 1.
         X7 = .8
        call OUTFLOW (P_end(4), Aet)
         U4(2) = -U4(2)
       else
        call INFLOW (P_end(4), Aet)
       end if
c
      end if
c
c  Converged properties are added to global solution matrix
c
      P(1,j+1) = P4(2)
      U(1,j+1) = U4(2)
c
      return
      end
c
c*********************************************************************
c
      SUBROUTINE RIGHT (P_end, Aet, loc)
c
c This subroutine prepares the initial values for MOC at the right end
c
      double precision gamma, dt, eps, P, U, X, X6, X7, P6, P7, U6, U7,
     &      X2, P4, U4, P_end(4), Aet
      integer loc, nx, j
c
      common/aa/gamma, dt, eps, nx, j
      common/cc/P(11,500), U(11,500), X(11)
      common/dd/X6, X7, P6, P7, U6, U7, X2(2), P4(2), U4(2)
c
c  Initial value line established for MOC
c
      X6 = X(nx)
      X7 = X(nx-1)
      P6 = P(nx,j)
      P7 = P(nx-1,j)
      U6 = U(nx,j)
      U7 = U(nx-1,j)
c
c  Solution points are given initial guess
c
      X2(1) = X7
      X4 = X6
```

```
      P4(1) = P6
      U4(1) = U6
c
c  Proper boundary condition is called
c
      if (Aet .eq. 0.) then
        U4(1) = 0.
       call CLOSED_R
      else if (loc .eq. 0) then
        if (U(nx,j) .lt. 0.) then
          U4(1) = -U4(1)
          U6 = -U6
          U7 = -U7
          X6 = 0.
          X7 = .2
         call INFLOW (P_end(2), Aet)
          U4(2) = -U4(2)
        else
          P4(1) = P_end(2)
         call OUTFLOW (P_end(2), Aet)
        end if
c
     else if (loc .gt. 0) then
       P4(1) = P_end(3)
      call OUTFLOW (P_end(3), Aet)
     end if
c
c  Converged properties are added to global solution matrix
c
     P(nx,j+1) = P4(2)
     U(nx,j+1) = U4(2)
c
     return
     end
c
c*********************************************************************
c
     SUBROUTINE CLOSED_L
c
c  This subroutine performs implicit predictor-corrector to calculate
c   properties at the left end when it is closed.
c
      double precision gamma, dt, eps, X6, X7, P6, P7, U6, U7, X2, P2,
    &        P4, U2, U4, A2, A4
     integer j, conv, nx
c
     common/aa/gamma, dt, eps, nx, j
      common/dd/X6, X7, P6, P7, U6, U7, X2(2), P4(2), U4(2)
c
     conv = 1
c
     do 100 while (conv .ne. 0)
c
c  Properties are linearly interpolated where the characteristic
```

```
c    intersects the initial value line
c
       U2 = U6 + (U7 - U6) * (X2(1) - X6) / (X7 - X6)
       P2 = P6 + (P7 - P6) * (X2(1) - X6) / (X7 - X6)
c
c  Speed of sound is calculated
c
       A2 = P2**((gamma - 1.) / (2. * gamma))
       A4 = A2 - (gamma - 1.) / 2. * (U2 - U4(1))
c
c  Corrector values calculated by MOC
c
       X2(2) = X6 - dt * (U2 - A2 + U4(1) - A4) / 2.
       P4(2) = A4**(2. * gamma / (gamma - 1.))
c
c  Convergence of solution is checked
c
       conv = 0
       call CONVERGE (P4(2), P4(1), eps, conv)
       call CONVERGE (X2(2), X2(1), eps, conv)
c
c  Solution point values are updated
c
       X2(1) = X2(2)
       P4(1) = P4(2)
c
100   continue
c
c  Velocity if updated
c
      U4(2) = 0.
c
      return
      end
c
c*********************************************************************
c
      SUBROUTINE INFLOW (P_end, Aet)
c
c  This subroutine performs implicit predictor-corrector MOC to calculate
c    the properties of inflow at the left end
c
      double precision gamma, P_end, A_end, dt, X6, X7, P6, P7, U6, U7,
     &        i, A2, A4, U2, U4, P2, P4, X2, X4, eps, Aet
      integer j, conv, nx
      common/aa/gamma, dt, eps, nx, j
      common/dd/X6, X7, P6, P7, U6, U7, X2(2), P4(2), U4(2)
c
c  The convergence criteria is initally set for an unconverged solution
c
      conv = 1
      i = 0.
c
c  The external speed of sound is calculated
```

```
c
      A_end = P_end**((gamma - 1.) / (2. * gamma))
c
c  Iterative loop proceeds until solution meets convergence criteria
c
      do 100 while (conv .ne. 0)
c
c  Counter
c
        i = i + 1.
c
c  Properties are linearly interpolated where the characteristic
c    intersects the initial value line
c
        U2 = U6 + (U7 - U6) * (X2(1) - X6) / (X7 - X6)
        P2 = P6 + (P7 - P6) * (X2(1) - X6) / (X7 - X6)
c
c  Speed of sound calculated
c
        A2 = P2**((gamma - 1.) / (2. * gamma))
c
c  Inflow through a partially open duct is considered
c
        if (Aet .gt. 1.) then
c
c  Speed of sound calculated based on the characteristic equation
c
          A4 = A2 + (gamma - 1.) / 2. * (U4(1) - U2)
c
c  Exit flow for an isentropic cd-nozzle is determined
c
          call  CD_NOZZLE (gamma, P_end, Aet, P4(1), U4(2))
c
c  Inflow through a fully open duct is considered
c
        else
c
c  Bournulli's equation applies
c
          A4 = dsqrt(A_end**2 - (gamma - 1.) / 2. * U4(1)**2)
c
c  Inlet velocity is calculated based on the characteristic equation
c
          U4(2) = U2 + 2. / (gamma - 1.) * (A4 - A2)
c
        end if
c
c  Corrector solution points are calculated by MOC
c
        X2(2) = X6 - dt * (U2 + U4(1) - A2 - A4) / 2.
        P4(2) = A4**(2. * gamma / (gamma - 1.))
c
c  Convergence of solution is checked
c
```

```
      conv = 0
      call CONVERGE (P4(2), P4(1), eps, conv)
      call CONVERGE (U4(2), U4(1), eps, conv)
      call CONVERGE (X2(2), X2(1), eps, conv)
c
c  Solution values are updated
c
c       X2(1) = (24. * X2(1) + 1. * X2(2)) / 25.
c       U4(1) = (24. * U4(1) + 1. * U4(2)) / 25.
c       P4(1) = (24. * P4(1) + 1. * P4(2)) / 25.
      X2(1) = (dsqrt(i-1) * X2(1) + 1. * X2(2)) / (1. + dsqrt
     &      (i-1))
      U4(1) = (dsqrt(i-1) * U4(1) + 1. * U4(2)) / (1. + dsqrt
     &      (i-1))
      P4(1) = (dsqrt(i-1) * P4(1) + 1. * P4(2)) / (1. + dsqrt
     &      (i-1))
c       write(*,*) i, X2(1), U4(1), P4(1)
c      pause
c
100   continue
c
      return
      end
c
c**********************************************************************
c
      SUBROUTINE INTERIOR (c)
c
c This subroutine performs implicit predictor-corrector MOC to calculate
c  the interior properties using MOC
c
      double precision gamma, dt, P, U, X, P5, P6, P7, U5, U6, U7, X5,
     &      X6, X7, X1(2), X2(2), X4, A1, A2, A4, P1, P2, P4(2), U1,
     &      U2, U4(2), Qm, Qp, eps
      integer c, j, conv, nx
      common/aa/gamma, dt, eps, nx, j
      common/cc/P(11,500), U(11,500), X(11)
c
c  The convergence criteria is initally set for an unconverged solution
c
      conv = 1
c
c  Initial value line established for MOC
c
      X5 = X(c-1)
      P5 = P(c-1,j)
      U5 = U(c-1,j)
      X6 = X(c)
      U6 = U(c,j)
      P6 = P(c,j)
      X7 = X(c+1)
      U7 = U(c+1,j)
      P7 = P(c+1,j)
c
```

```
c  Solution points given initial values
c
      X1(1) = X5
      X2(1) = X7
      X4 = X6
      P4(1) = P6
      U4(1) = U6
c
c  Iterative loop proceeds until solution meets convergence criteria
c
      do 100 while (conv .ne. 0)
c
c  Properties are linearly interpolated where the characteristics
c    intersects the initial value line
c
        U1 = U6 + (U5 - U6) * (X1(1) - X6) / (X5 - X6)
        P1 = P6 + (P5 - P6) * (X1(1) - X6) / (X5 - X6)
        U2 = U6 + (U7 - U6) * (X2(1) - X6) / (X7 - X6)
        P2 = P6 + (P7 - P6) * (X2(1) - X6) / (X7 - X6)
c
c  Speed of sound is calculated
c
        A1 = P1**((gamma - 1.) / (2. * gamma))
        A2 = P2**((gamma - 1.) / (2. * gamma))
        A4 = P4(1)**((gamma - 1.) / (2. * gamma))
c
c  Characteristic equations solved
c
        X1(2) = X6 - DT * (U1 + U4(1) + A1 + A4) / 2.
        X2(2) = X6 - DT * (U2 + U4(1) - A2 - A4) / 2.
c
c  Compatibility equations solved
c
        Qp = gamma * ((P1 + P4(1)) / 2.)**((gamma + 1.) / (2. * gamma))
        Qm = gamma * ((P2 + P4(1)) / 2.)**((gamma + 1.) / (2. * gamma))
c
c  Corrector solution points are calculated by MOC
c
        U4(2) = ((P2 - Qm * U2) - (P1 + Qp * U1)) / (-Qm - Qp)
        P4(2) = ((P1 + Qp * U1) * (-Qm) - (P2 - Qm * U2) * Qp) /
     &    (-Qm - Qp)
c
c  Convergence of solution is checked
c
        conv = 0
        call CONVERGE (P4(2), P4(1), eps, conv)
        call CONVERGE (U4(2), U4(1), eps, conv)
        call CONVERGE (X1(2), X1(1), eps, conv)
        call CONVERGE (X2(2), X2(1), eps, conv)
c
c  Solution values are updated
c
        X1(1) = X1(2)
        X2(1) = X2(2)
```

```
        P4(1) = P4(2)
        U4(1) = U4(2)
c
 100  continue
c
c  Converged properties are added to global solution matrix
c
      U(c,j+1) = U4(2)
      P(c,j+1) = P4(2)
c
      return
      end
c
c**********************************************************************
c
      SUBROUTINE OUTFLOW (P_end, Aet)
c
c  This subroutine performs implicit predictor-corrector MOC to calculate
c   subsonic flow properties at the right end when it is open
c
      double precision gamma, P_end, dt, P6, P7, U6, U7, X6, X7, X2,
     &       i, X4, P2, P4, U2, U4, A2, A4, eps, Aet
      integer j, conv, nx
      common/aa/gamma, dt, eps, nx, j
      common/dd/X6, X7, P6, P7, U6, U7, X2(2), P4(2), U4(2)
c
c  The convergence criteria is initally set for an unconverged solution
c
      conv = 1
      i = 0.
c
      do 100 while (conv .ne. 0)
c
c  Counter
c
       i = i + 1.
c
c  Properties are linearly interpolated where the characteristic
c   intersects the initial value line
c
       U2 = U6 + (U7 - U6) * (X2(1) - X6) / (X7 - X6)
       P2 = P6 + (P7 - P6) * (X2(1) - X6) / (X7 - X6)
c
c  Speed of sound calculated
c
       A2 = P2**((gamma - 1.) / (2. * gamma))
       A4 = P4(1)**((gamma - 1.) / (2. * gamma))
c
c  Outflow through a partially open duct is modeled as a converging
c   nozzle where the end of the duct is the nozzle inlet
c
       if (Aet .gt. 1.) then
        call C_NOZZLE (gamma, P_end, Aet, P4(1), U4)
c
```

```
c  Outflow through a fully open duct is considered using the
c    characteristic equation
c
        else
          U4(2) = -(A4 - A2) * 2. / (gamma - 1.) + U2
          P4(2) = P4(1)
        end if
c
c  Corrector values calculated by MOC
c
        X2(2) = X6 - dt * (U2 + A2 + U4(1) + A4) / 2.
        A4 = A2 - (U4(2) - U2) * (gamma - 1.) / 2.
c
c  For the partially open duct, the speed of sound is calculated
c
        if (Aet .gt. 1.) then
          P4(2) = A4**((2. * gamma) / (gamma - 1.))
        end if
c
c  Convergence of solution is checked
c
        conv = 0
        call CONVERGE (P4(2), P4(1), eps, conv)
        call CONVERGE (U4(2), U4(1), eps, conv)
        call CONVERGE (X2(2), X2(1), eps, conv)
c
c  Solution point values are updated
c
c         X2(1) = (dsqrt(i-1) * X2(1) + 1. * X2(2)) / (1. + dsqrt
c    &        (i-1))
c         U4(1) = (dsqrt(i-1) * U4(1) + 1. * U4(2)) / (1. + dsqrt
c    &        (i-1))
c         P4(1) = (dsqrt(i-1) * P4(1) + 1. * P4(2)) / (1. + dsqrt
c    &        (i-1))
        X2(1) = (X2(2) + 9. * X2(1)) / 10.
        U4(1) = (U4(2) + 9. * U4(1)) / 10.
        P4(1) = (P4(2) + 9. * P4(1)) / 10.
c         X2(1) = (dsqrt(i) * X2(1) + 1. * X2(2)) / (1. + dsqrt(i))
c         U4(1) = (dsqrt(i) * U4(1) + 1. * U4(2)) / (1. + dsqrt(i))
c         P4(1) = (dsqrt(i) * P4(1) + 1. * P4(2)) / (1. + dsqrt(i))
c
100   continue
c
c  If the outflow velocity is supersonic, SONIC is called
c
        if (abs(U4(2)) .gt. A4) then
c         call SONIC (P_end)
          write(*,*) 'Outflow is sonic'
c          write(*,*) P4(1), U4(1)
c         pause
        end if
c
      return
      end
```

```
c
c**********************************************************************
c
     SUBROUTINE SONIC (P_end)
c
c This subroutine performs implicit predictor-corrector MOC to calculate
c  properties for choked flow exiting the right end.  Because of the
c  constant area duct, the flow cannot become supersonic.
c
      double precision gamma, P_end, dt, P, U, X, P6, P7, U6, U7, X6,
     &       X7, X2, X4, P2, P4, U2, U4, A2, A4, eps
     integer j, conv, nx
     common/aa/gamma, dt, eps, nx, j
      common/dd/X6, X7, P6, P7, U6, U7, X2(2), P4(2), U4(2)
c
c  The convergence criteria is initally set for an unconverged solution
c
     conv = 1
c
c  Solution points given initial guess
c
     P4(1) = P_end
c
c  Iterative loop proceeds until solution meets convergence criteria
c
     do 100 while (conv .ne. 0)
c
c  Properties are linearly interpolated where the characteristic
c   intersects the initial value line
c
      U2 = U6 + (U7 - U6) * (X2(1) - X6) / (X7 - X6)
      P2 = P6 + (P7 - P6) * (X2(1) - X6) / (X7 - X6)
c
c  Speed of sound calculated
c
      A2 = P2**((gamma - 1.) / (2. * gamma))
      A4 = P4(1)**((gamma - 1.) / (2. * gamma))
c
c  Corrector values calculated by MOC
c

      X2(2) = X6 - dt * (U2 + A2 + U4(1) + A4) / 2.
      U4(2) = (2. * A2 - (gamma - 1.) * U2) / (gamma + 1.)
      A4 = abs(U4(2))
      P4(1) = A4**(2. * gamma / (gamma - 1.))
c
c  Convergence of solution is checked
c
      conv = 0
      call CONVERGE (U4(2), U4(1), eps, conv)
      call CONVERGE (X2(2), X2(1), eps, conv)
c
c  Solution point values are updated
c
```

```
      X2(1) = X2(2)
      U4(1) = U4(2)
c
100   continue
c
      return
      end
c
c*********************************************************************
c
      SUBROUTINE CLOSED_R
c
c This subroutine performs implicit predictor-corrector at the right end
c   when it is closed
c
      double precision gamma, dt, P, U, X, P6, P7, U6, U7, X6, X7,
     &       X2, X4, P2, P4, U2, U4, A2, A4, eps
      integer j, conv, nx
      common/aa/gamma, dt, eps, nx, j
      common/dd/X6, X7, P6, P7, U6, U7, X2(2), P4(2), U4(2)
c
c The convergence criteria is initally set for an unconverged solution
c
      conv = 1
c
      do 100 while (conv .ne. 0)
c
c Properties are linearly interpolated where the characteristic
c   intersects the initial value line
c
      U2 = U6 + (U7 - U6) * (X2(1) - X6) / (X7 - X6)
      P2 = P6 + (P7 - P6) * (X2(1) - X6) / (X7 - X6)
c
c Speed of sound calculated
c
      A2 = P2**((gamma - 1.) / (2. * gamma))
      A4 = A2 + (gamma - 1.) / 2. * (U2 - U4(1))
c
c Corrector solution points are calculated using MOC
c
      X2(2) = X6 - dt * (U2 + A2 + U4(1) + A4) / 2.
      P4(2) = A4**(2. * gamma / (gamma - 1.))
c
c Convergence of solution is checked
c
      conv = 0
      call CONVERGE (P4(2), P4(1), eps, conv)
      call CONVERGE (X2(2), X2(1), eps, conv)
c
c Solution values are updated
c
      X2(1) = X2(2)
      P4(1) = P4(2)
c
```

```
100  continue
c
c  Velocity is updated
c
     U4(2) = 0.
c
     return
     end
c
c*********************************************************************
c
     SUBROUTINE C_NOZZLE (gamma, P_end, Ait, Pi, Ui)
c
c  This subroutine controls the openning and closing of the ports.
c
     double precision gamma, P_end, P0i, Pi, Ui(2), Ai, Mi, Mt, Ait,
    &       Att, Aii, Aq, div
c
c  The speed of sound at the inlet is calculated
c
     Ai = Pi**((gamma - 1.) / (2. * gamma))
c
c  Mach # at the inlet is calculated
c
     Mi = abs(Ui(1)) / Ai
c
c  The inlet stagnation pressure is determined
c
     P0i = Pi * (1. + (gamma - 1.) / 2. * Mi**2)**(gamma / (gamma -
    &   1.))
c
c  The stagnation pressure (P0i) must be greater than the back pressure
c   (P_end) for outflow to occur
c
     if (P0i .le. P_end) then
       P0i = 1.005 * P_end
     end if
c
c  Mach # at the throat is calculated based on back pressure (P_end)
c   and inlet stagnation pressure (P0i)
c
     Mt = dsqrt(2. / (gamma - 1.) * ((P0i / P_end)**((gamma - 1.) /
    &   gamma) - 1.))
c
c  When the throat becomes choked, the throat pressure is independent
c   of the ambient back pressure (P_end) and Mt = 1
c
     if (Mt .gt. 1.) then
       Mt = 1.
     end if
c
c  The area ratios for throat-to-choked and inlet-to-choked are calculated
c
     Att = 1./Mt*(2./(gamma+1.)*(1.+(gamma-1.)/2.*Mt**2))**((gamma
```

```
     &     +1.)/(2.*(gamma-1.)))
     Aii = Ait * Att
c
c  The Mach # (Mi) corresponding to the inlet area ratio is determined by
c    iterating Mi until the calculated area ratio (Aq) equals that actual
c    area ratio
c
     Mi = 0.
     div = .1
     do 110 while (div .ge. .1**10)
       Aq = 999.
       do 111 while (Aq .ge. Aii)
         Mi = Mi + div
         Aq = 1. / Mi * (2. / (gamma + 1.) * (1. + (gamma - 1.) / 2.
    &        * Mi**2))**((gamma + 1.) / (2. * (gamma - 1.)))
111     continue
       Mi = Mi - div
       div = div * .1
110   continue
c
c  The nozzle inlet velocity is calculated
c
     Ui(2) = Mi * Ai
c
     return
     end
c
c*********************************************************************
c
     SUBROUTINE CD_NOZZLE (gamma, P_end, Aet, Pe, Ue)
c
c  This subroutine calculates the exit velocity for a converging-diverging
c    nozzle.
c
     double precision gamma, P_end, Pe, P_sub, P_sup, Pq, Pu, P_ns,
    &       Aet, Aq, Me, Me_sub, Me_sup, Ae, Ue, div
c
c  The Mach # (Me_sub) corresponding to the choking point of the nozzle
c    is determined by iterating Me_sub until the calculated area ratio
c    (Aq) equals that actual area ratio (Aet)
c
      Me_sub = 0.
      div = .1
      do 110 while (div .ge. .1**10)
        Aq = 999.
        do 111 while (Aq .ge. Aet .and. Me_sub .lt. 1.)
          Me_sub = Me_sub + div
          Aq = 1. / Me_sub * (2. / (gamma + 1.) * (1. + (gamma - 1.) /
    &        2. * Me_sub**2))**((gamma + 1.) / (2. * (gamma - 1.)))
111       continue
        Me_sub = Me_sub - div
        div = div * .1
110    continue
c
```

```
c  The back pressure (P_sub) at the choking point is determined
c
      P_sub = P_end * (1. + (gamma - 1.) / 2. * Me_sub**2)**(-gamma /
     &       (gamma-1))
c
c  The Mach # (Me_sup) corresponding to the supersonic design point of
c   the nozzle is determined by iterating Me_supuntil the calculated
c   area ratio (Aq) equals that actual area ratio (Aet)
c
      Me_sup = 0.
      div = 1.
      do 120 while (div .ge. .1**10)
       Aq = 1.
       do 121 while (Aq .le. Aet)
        Me_sup = Me_sup + div
        Aq = 1. / Me_sup * (2. / (gamma + 1.) * (1. + (gamma - 1.) /
     &       2. * Me_sup**2))**((gamma + 1.) / (2. * (gamma - 1.)))
121     continue
       Me_sup = Me_sup - div
       div = div * .1
120   continue
c
c  The back pressure (P_sup) at the supersonic design point is determined
c
      P_sup = P_end * (1. + (gamma - 1.) / 2. * Me_sup**2)**(-gamma /
     &       (gamma-1))
c
c  The back pressure (P_ns) corresponding to a normal shock located at
c   the nozzle exit is calculated.  Pu is the pressure upstream of the
c   normal shock
c
      Pu = P_end * (1. + (gamma - 1.) / 2. * Me_sup**2)**(-gamma /
     &    (gamma - 1.))
      P_ns = Pu * (1.+ 2. * gamma / (gamma + 1.) * (Me_sup**2 - 1.))
c
c  When the exit pressure is greater than the choking point pressure, the
c   flow at the exit is subsonic.  The exit Mach # is iterated until the
c   calculated pressure equals the actual exit pressure
c
      If (Pe .ge. P_sub) then
       Me = 0.
       div = .1
       do 130 while (div .ge. .1**10)
        Pq = 999.
        do 131 while (Pq .ge. Pe)
         Me = Me + div
         Pq = P_end * (1. + (gamma - 1.) / 2. * Me**2)**(-gamma /
     &        (gamma - 1.))
131      continue
        Me = Me - div
        div = div * .1
130    continue
c
c  When the exit pressure lies between the choking point pressure and the
```

```
c    exit normal shock pressure, a normal shock exists within the nozzle.
c    The exit Mach # is iterated until the calculated pressure equals the
c    actual exit pressure
c
       else if (Pe .lt. P_sub .and. Pe .ge. P_ns) then
        Me = 0.
        div = .1
        do 140 while (div .ge. .1**10)
         Pq = 999.
         do 141 while (Pq .ge. Pe)
          Me = Me + div
          Pq = P_end / Aet * (1. + (gamma - 1.) / 2. * Me**2)**
     &        (-gamma / (gamma - 1.)) / Me * (2. / (gamma + 1. ) *
     &        (1. + (gamma - 1.) / 2. * Me**2))**((gamma + 1.) / (2.
     &        * (gamma - 1.)))
141        continue
        Me = Me - div
        div = div * .1
140     continue
c
c   When the exit pressure is less than the exit normal shock pressure,
c    the exit flow is supersonic.  This exit condition is not considered
c
      else
c        write(*,*) 'The flow is supersonic at the cd-nozzle exit'
c        write(*,*) Aet
        Pe = P_sup
        Me = Me_sup
c        pause
      end if
c
c   The speed of sound at the nozzle exit is found
c
      Ae = Pe**((gamma - 1.) / (2. * gamma))
c
c   The flow velocity at the nozzle exit is calculated
c
      Ue = Me * Ae
c
     return
     end
c
c**********************************************************************
c
     SUBROUTINE PORT (P_end, part, loc, cp, nj, next, rate, Aet)
c
c   This subroutine determines the opening and closing of the ports based
c    on flow conditions.
c
     double precision gamma, P_end(4), Y_l, Y_h, P, U, X, mass, eps,
    &      dt, rate, Aet(2)
     integer conv, i, j, nj, loc(4), part(2), dt_l, dt_r, Pt, next,
    &      cp(4), flag
c
```

```
      common/aa/gamma, dt, eps, nx, j
      common/bb/Y_l(500), Y_h(500), mass(500), Pt(4,2)
      common/cc/P(11,500), U(11,500), X(11)
c
c  The number of time steps since opening or closing are tracked
c
      part(1) = part(1) + 1
      part(2) = part(2) + 1
c
      if (flag .eq. 0) then
c
c  For the first time step, the high pressure inflow port opens
c    (location #1) and the end resets to begin gradual opening
c
      if (j .eq. 1) then
       loc(1) = 1
       part(1) = 1
c
c  While high pressure gas is flowing in, the pressure at the outflow
c    end is analyzed for greater internal pressure.  At that time, the
c    high pressure outflow duct is opens (location #2) and the partial
c    opening right end is reset
c
      else if (loc(2) .eq. 0) then
c        write(*,*) "Port 1 is open.  Ports 2, 3 & 4 are closed."
c
       if (P(nx,j) .ge. P_end(2)) then
        loc(2) = 1
        part(2) = 1
        Pt(1,2) = j
       end if
c
c  While both high pressure ports are open, the pressure at the inflow
c    end is analyzed for the pressure rising above ambient pressure.
c    The high pressure inflow port (location #3) is closed
c
      else if (loc(1) .eq. 1 .or. loc(2) .eq. 1) then
c        write(*,*) "Ports 1 & 2 are open.  Ports 3 & 4 are closed."
c
       if (loc(1) .eq. 1) then
         if (P(1,j) .ge. P_end(1) .or. U(1,j) .lt. 0. .or. j .eq.
     &     cp(1)) then
          loc(1) = 2
          part(1) = 1
          Pt(2,1) = j
        end if
       end if
c
c  The high pressure outflow port is closed (location #4) when the
c    interphase line reached the right end of the duct or the interphase
c    reverses direction flowing back to the left
c
c     else if (loc(1) .eq. 2 .and. loc(2) .eq. 1) then
c       write(*,*) "Port 2 is open.  Ports 1, 3 & 4 are closed."
```

```
c
        if (loc(2) .eq. 1) then
          if (Y_h(j) .ge. 1. .or. Y_h(j) .lt. Y_h(j-1) .or. j .eq.
     &      cp(2)) then
           if (Y_h(j) .ge. 1.) then
            Y_h(j) = 1.
           end if
           loc(2) = 2
           part(2) = 1
           Pt(2,2) = j
         end if
        end if
c
      else if (loc(1) .eq. 2 .and. loc(2) .eq. 2 .and. Aet(1) .eq. 0.
     & .and. Aet(2) .eq. 0.) then
        flag = 1
      end if
c
c
      else if (flag .eq. 1) then
c
c  At this point all ports are closed.  The low pressure outflow port
c   opens (location #5) when the flow velocity near the right end reaches
c   a maximum.  The kinetic energy of the flow is best utilized at this
c   time
c
      if (loc(3) .eq. 0 .and. loc(4) .eq. 0) then
c       write(*,*) "All ports are closed."
c
        if (U(nx-1,j) - U(nx-1,j-1) .le. 0. .and. U(nx-1,j-1) - U(nx-1,
     &   j-2) .ge. 0.) then
          j = j - 1
          loc(3) = 1
          part(2) = 1
          Pt(3,2) = j
        end if
c
c  Once the pressure at the left end of the duct drops below the external
c   pressure at left, the low pressure inflow port opens (location #6)
c
      else if (loc(3) .eq. 1 .and. loc(4) .eq. 0) then
c       write(*,*) "Port 3 is open.  Ports 1, 2 & 4 are closed."
c
        if (P(1,j) .le. P_end(4)) then
          loc(4) = 1
          part(1) = 1
          Pt(3,1) = j
        end if
c
c  The low pressure outflow port closes (location #7) after the interphase
c   line reaches the right end of the duct
c
      else if (loc(3) .eq. 1) then
c       write(*,*) "Ports 3 & 4 are open.  Ports 1 & 2 are closed."
```

```
c
       if (Y_l(j) .ge. 1. .or. Y_l(j) .lt. Y_l(j-1) .or. j .eq. cp(3))
     &   then
        if (Y_l(j) .ge. 1.) then
         Y_l(j) = 1.
        end if
        loc(3) = 2
        part(2) = 1
        Pt(4,2) = j
       end if
c
c  The low pressure inflow port closes (location #8) when the internal
c    pressure is greater than the external pressure thus transitioning
c    from inflow to outflow
c
      else if (loc(4) .eq. 1) then
c        write(*,*) "Port 4 is open.  Ports 1, 2 & 3 are closed."
c
       if (P(1,j) .ge. P_end(4) .or. U(1,j) .le. 0. .or. j .eq. cp(4))
     &   then
        loc(4) = 2
        part(1) = 1
        Pt(4,1) = j
       end if
c
c  At this point all ports are closed.  The high pressure inflow port
c    opens (location #1) when the velocity near the port reachs a
c    maximum.  This completes a cycle
c
      else if (loc(3) .eq. 2 .and. loc(4) .eq. 2 .and. Aet(1) .eq. 0.
     & .and. Aet(2) .eq. 0.) then
       flag = 2
      end if
c
c
      else if (flag .eq. 2) then
c        write(*,*) "All ports are closed."
c
       if (U(2,j) - U(2,j-1) .le. 0. .and. U(2,j-1) - U(2,j-2) .ge. 0.)
     &   then
        j = j - 1
        conv = 0
c
c  The initial and final flow values are checked for convergence
c
        do 101 k = 1, nx
         call converge(P(k,1), P(k,j), eps, conv)
         call converge(U(k,1), U(k,j), eps, conv)
101     continue
c
c  Once convergence is acheived, the calculated values are recorded
c
        if (conv .eq. 0) then
         call OUTPUT
```

```
        end if
c
c  Otherwise, the cycle restarts at the beginning after updating the
c    initial values
c
        loc = 0
        part(1) = 1
        Pt(1,1) = j
        flag = 0
c
        do 100 i = 1, nx
c           P(i,1) = (P(i,1) + P(i,j)) / 2.
c           U(i,1) = (U(i,1) + U(i,j)) / 2.
          P(i,1) = P(i,j)
          U(i,1) = U(i,j)
100       continue
c
        next = 1
      end if
c
    end if
c
c  Subroutine called which calculates the how much the port is open.
c
    call PARTIAL (rate, Aet, part, loc)
c
    return
    end
c
c*********************************************************************
c
    SUBROUTINE PARTIAL (rate, Aet, part, loc)
c
c  This subroutine calculates the area ratio (duct / throat) for a
c    gradually opening port.
c
    double precision rate, Aet(2)
    integer part(2), loc(4)
c
c  The opening of the left end port is determined
c
    if (loc(1) .eq. 1 .or. loc(4) .eq. 1) then
      call OPEN (rate, Aet(1), part(1))
    else
      call CLOSE (rate, Aet(1), part(1))
    end if
c
c  The opening of the right end port is determined
c
    if (loc(2) .eq. 1 .or. loc(3) .eq. 1) then
      call OPEN (rate, Aet(2), part(2))
    else
      call CLOSE (rate, Aet(2), part(2))
    end if
```

```
c
      return
      end
c
c***********************************************************************
c
      SUBROUTINE OPEN (rate, A, part)
c
c  This subroutine calculates the area ratio (duct / throat) for a
c   gradually opening port.
c
      double precision rate, A
      integer part
c
c  The area ratio is based on the number of time steps since opening
c   began and the rate the port opens during each time step
c
      A = 1. / (part * rate)
c
c  The area ratio must be greater than (partially open) or equal to 1
c   (fully open)
c
      If (A .lt. 1.) then
        A = 1.
      end if
c
      return
      end
c
c***********************************************************************
c
      SUBROUTINE CLOSE (rate, A, part)
c
c  This subroutine calculates the area ratio (duct / throat) for a
c   gradually closing port.
c
      double precision rate, A
      integer part
c
c  The area ratio is based on the number of time steps since closing
c   began and the rate the port opens during each time step
c
      A = 1. / (1. - part * rate)
c
c  For a large area ratio, the port is assumed closed
c
      if (A .gt. 500. .or. A .lt. 0.) then
        A = 0.
      end if
c
      return
      end
c
c***********************************************************************
```

```
c
      SUBROUTINE MASSFLOW (mass)
c
c  This subroutine calculates the mass flowing in through the low pressure
c    port.
c
      double precision gamma, dt, eps, mass(500), P, U, X, rho, dmass
      integer j, nx
      common/aa/gamma, dt, eps, nx, j
      common/cc/P(11,500), U(11,500), X(11)
c
c  The density of the entering gas is calculated
c
      rho = P(1,j)**(1./gamma)
c       rho = P(nx,j)**(1./gamma)
c
c  The change in mass flux from the current time step to the future time
c    step is determined
c
      dmass = rho * U(1,j) * dt
c       dmass = rho * U(nx,j) * dt
c
c  The mass flux at the future time step is calculated
c
      mass(j+1) = mass(j) + dmass
c
      return
      end
c
c**********************************************************************
c
      SUBROUTINE DISTANCE (Y, U)
c
c  This subroutine calculates the position of the interphase line.
c
      double precision gamma, U(11, 500), Y(500), U_avg(500), dt, eps
      integer c, j, nx
      common/aa/gamma, dt, eps, nx, j
c
c  This do-loop determines which station is located to the left of the
c    interphase line
c
      do 100 c = 2, nx
        if (Y(j) .ge. (c - 2.) / (nx - 1.) .and. Y(j) .lt. (c - 1.) /
     &     (nx - 1.)) then
          goto 101
        end if
100   continue
101   continue
c
c  The average velocity at the location of the interphase line is
c    calculated
c
        U_avg(j) = U(c-1,j) + ((U(c,j) - U(c-1,j)) * (Y(j) * (nx - 1.)
```

```
      &        - (c - 2.)))
c
c  The future location of the interphase line is determined
c
       Y(j+1) = Y(j) + U_avg(j) * dt
c
c  The interphase line must exist inside the tube
c
       if (Y(j+1) .lt. 0.) then
         Y(j+1) = 0.
       else if (Y(j+1) .gt. 1. .or. Y(j) .ge. 1.) then
         Y(j+1) = 1.
       end if
c
      return
      end
c
c**********************************************************************
c
      SUBROUTINE CONVERGE (z1, z2, eps, conv)
c
c This subroutine determines the convergence of an iterated variable
c   based on percent change
c
      double precision z1, z2, eps, num
      integer conv
c
c  The percent change of values is calculated
c
      num = 1. - z1 / z2
c
c  The percent change is checked against the convergence criteria
c
      if (abs(num) .ge. eps) then
        conv = 1
      end if
c
      return
      end
c
c**********************************************************************
```

```
10  LPRINT "Program JUNE252I.BAS, HHK 06/25/1995"
20  PRINT "Program for BASICA, limited to 200 timesteps"
30  LPRINT DATE$, TIME$
50  LPRINT
60  LPRINT "Onedimensional Non-steady Homentropic Flow"
70  LPRINT "Inverse Marching Euler (Implicit) Predictor-Corrector Method"

80  LPRINT "Modified 12/2/86 by HHK, from Zucrow and Hoffman GASDYNAMICS"
    90      LPRINT "Program handles gradual opening of pipe end on left, subsonic and
    critical outflow, and inflow, or left and closed. Right end initially closed, but
    REEDVALVE allows (subsonic) inflow from selected external pressure levels";
100 LPRINT "with plot for pressure and velocity vs time for selected stations,HHK12/20/86"
110 DIM X(21), P(21, 200), U(21, 200), X1(21), X2(21), U4(21), P4(21), A4(21),
    P1(21), P2(21), U1(21), U2(21), A1(21), A2(21), DIST(200), Y(200), UAV(200),
    MASS(200)
120 DEFINT C, J, L, N
130 PRINT "enter GAMMA=";
140 INPUT G0
150 LPRINT "Gamma="; G0
160 PRINT "enter (SI Units), Gas Constant, Stagn. Temp. (abs), Ref. Pressures P0 (air in
    right,gas out left), PA (discharge,air,left), PR (high press. gas in, right end)"
170 INPUT R0, T0, P00, PA0, PR0
180 RO = -1
190 LPRINT "Gas Constant R0="; R0; "Ref. Temp. (abs)="; T0; "Ref. Pressure,air in,right,
    gas out left ="; P00; "PA,discharge air left="; PA0, "PR,High pressure gas in ,right=";
    PR0, "(PRESSURES IN PASCALS)"
200 A0 = SQR(G0 * R0 * T0)
210 DEF FNA (P) = SQR(G0 * R0 * T0) * (P / P0) ^ ((G0 - 1) / (2 * G0))
220 DEF FNR (P) = P0 * (P / P0) ^ (1 / G0) / (R0 * T0)
230 DEF FNU (X) = (1 + (G0 - 1) * X * X / 2) ^ ((G0 + 1) / (2 * (G0 - 1))) / X
240 PRINT "Enter total Tube Length L (m), Number of Stations (incl. x=0 and x=L)"
250 INPUT L, NX
260 PRINT "Time Step delta t (sec) <"; L / ((NX - 1) * 2 * A0); " for numerical stability"
270 PRINT "Enter delta t (sec)=";
280 INPUT DT
290 LPRINT "delta t (sec)="; DT
300 PRINT "Enter # of Time Steps (<200)=";
310 INPUT NJ
320 LPRINT "Number of locations along X- AXIS ="; NX; "Number of time steps="; NJ
330 PRINT "Is an ideal REEDVALVE present at the right (closed) end (y/n)";
340 INPUT R$
350 IF R$ <> "y" THEN 370
360 LPRINT "Reedvalve at right end will be open as long as PR>P(NX,J)"
370 PRINT "Are Initial Conditions at t=0 uniform in Tube (y/n) ?"
380 INPUT U$
390 IF U$ = "y" THEN 500
400 FOR C = 1 TO NX
410 PRINT "c="; C, " at"; (C - 1) * L / (NX - 1); " (m), Enter U(c,1),P(c,1)="
```

```
420 INPUT U(C, 1), P(C, 1): X(C) = (C - 1) * L / (NX - 1)
430 UMIN = 2 * ((PA / P(1, 1)) ^ ((G0 - 1) / (2 * G0)) - 1) * A0 / (G0 - 1)
440 LPRINT X(C), U(C, 1), P(C, 1)
450 NEXT C
460 FOR J = 1 TO NJ
470 X(C) = X(C)
480 NEXT J
490 GOTO 600
500 PRINT "Enter U (m/sec), P (PA), const at t=0 in Tube";
510 INPUT UC, PC
520 LPRINT "Uniform Initial Conditions, U=";UC,"P=";PC
530 FOR C = 1 TO NX
540 U(C, 1) = UC: P(C, 1) = PC
550 X(C) = (C - 1) * L / (NX - 1)
560 FOR J = 1 TO NJ
570 X(C) = X(C)
580 NEXT J
590 NEXT C
600 PRINT "shall all data be printed? (y/n)"
610 INPUT B$
620 IF B$ = "y" THEN 690
630 LPRINT "intermediate data not printed"
640 PRINT "Enter TIME INTERVALLS for LEFT END OPEN or CLOSED (the latter will be
    recalculated, select it as a large value to be overridden)"
650 INPUT LEO, LEC
660 PRINT "enter the time for right end ready for AIR IN"
670 INPUT REA
680 LPRINT "Left End OPENS at"; LEO, "LEFT END CLOSES at"; LEC;"Right end ready
    for AIR IN at ";REA
685 P0=P00:PA=PA0:PR=PR0:COUNTER=0:FL1=0:FL2=0:FL3=0
686 FOR C=1 TO NX
687 LPRINT U(C,1),P(C,1)
688 NEXT
690 PRINT "Is Left End Closed INITIALLY? (y/n)";
700 INPUT Q$
710 IF Q$ = "n" THEN 770
720 LPRINT "Left End is INITIALLY CLOSED"
730 J = 1: GOSUB 3150
740 NO = 1
750 LPRINT "Left End Closed"
760 GOTO 940
770 PRINT "enter # of left end outflow steps =>1";
780 INPUT NO
790 LPRINT "Number of left end inflow steps="; NO
800 J = 1
810 C = 1
820 FOR J = 1 TO NO
830 PRINT "J="; J; " enter U(1,J)=>"; UMIN
840 IF J = 1 AND NO = 1 THEN 890
850 INPUT U(1, J): GOSUB 2580
860 LPRINT "J="; J; "U(1,J)="; U(1, J); "P(1,J)="; P(1, J)
870 NEXT J
880 GOTO 900
890 U(1, 1) = UMIN
```

```
900  FOR J = NO TO NJ
910  P(1, J) = P(1, NO): U(1, J) = U(1, NO)
920  NEXT J
930  MASS(1)=0
940  FOR J = 1 TO NJ
950  IF J <= NO THEN 1000
960  IF Q$ <>"y" THEN 990
970  GOSUB 3150
980  GOTO 1000
990  GOSUB 1150
1000 FOR C = 2 TO NX - 1
1010 GOSUB 1720
1020 NEXT C
1030 GOSUB 1980
1040 NEXT J
1050 LPRINT "Mass Outflows:="MASS(IJ+1)"at J="IJ+1;"H-P MASS,
     OUT="MASS(LEC+1)-MASS(IJ+1)
1060 LPRINT "Second Interphase Line reaches LEFT END at LEC=";LEC
1070 PRINT "Shall data file be saved (for LOTUS?) (y/n)";
1080 INPUT F$
1090 IF F$ = "y" THEN 2440
1100 PRINT "Pressure and Velocity Records for Selected Stations to be printed (y/n)?"
1110 INPUT C$
1120 IF C$ <> "y" THEN 3140
1130 GOTO 3010
1135 GOTO 4000
1140 END
1150 IF B$ = "n" THEN 1170
1160 PRINT "Subsonic flow, left end at station #1, x=0"
1170 X7 = X(2): P7 = P(2, J): U6 = U(1, J): X6 = 0: U7 = U(2, J): P6 = P(1, J)
1180 I = 1
1190 X2(1) = X7: P2(1) = P7: P4(1) = P(1, J): U4(1) = U6
1200 U2(I) = U6 + (U7 - U6) * (X2(I) - X6) / (X7 - X6)
1210 P2(I) = P6 + (P7 - P6) * (X2(I) - X6) / (X7 - X6)
1220 A2(I) = FNA(P2(I))
1230 A4(I) = FNA(P(1, J))
1240 X2(I + 1) = X6 - DT * (U2(I) + U4(I) - A2(I) - A4(I)) / 2
1250 U4(I + 1) = (FNA(P(1, J)) - FNA(P2(I))) * 2 / (G0 - 1) + U2(I)
1260 P4(I + 1) = PA
1270 P6 = PA
1280 P(1, J) = PA
1290 P4(I + 1) = P0 * (A4(I) / A0) ^ ((2 * G0) / (G0 - 1))
1300 P6 = PA
1310 IF ABS(U4(I + 1) - U4(I)) < .1 THEN 1350
1320 X2(I) = X2(I + 1): U4(I) = U4(I + 1)
1330 I = I + 1
1340 GOTO 1200
1350 IF B$ = "n" THEN 1370
1360 PRINT "U(1,J+1)="; U4(I + 1); "P(1,J+1)="; P4(1)
1370 IF U4(I + 1) > 0 THEN 1430
1380 IF B$ = "n" THEN 1400
1390 PRINT "Outflow"
1400 U(1, J + 1) = U4(I + 1): P(1, J + 1) = P4(1)
1410 IF ABS(U(1, J + 1)) >= FNA(P(1, J + 1)) THEN 3360
```

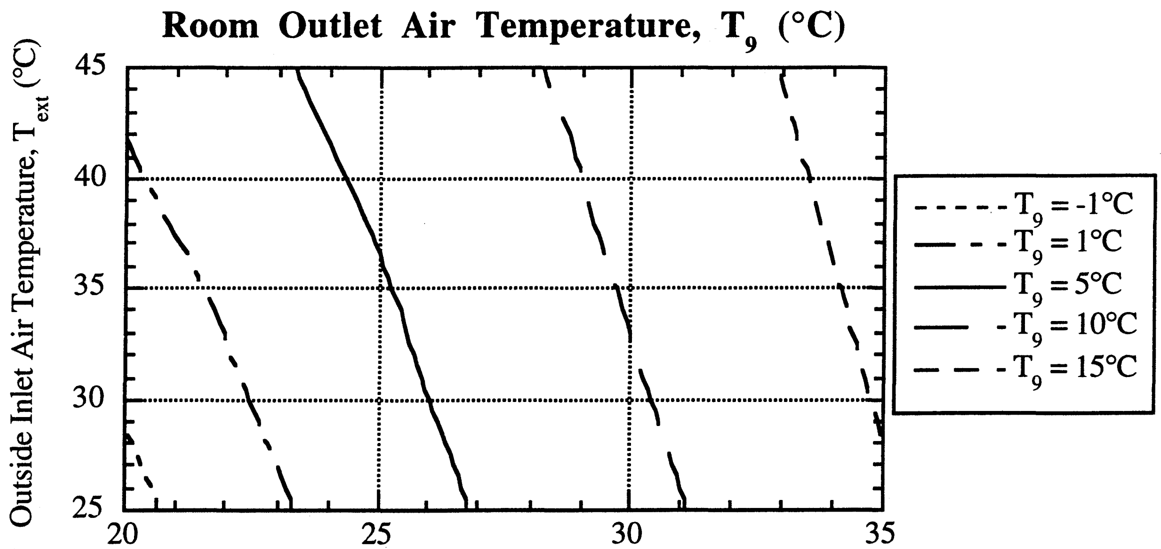
```
1420 RETURN
1430 IF B$ = "n" THEN 1450
1440 PRINT "Inflow at left end"
1450 I = 1
1460 X2(1) = X7: P2(1) = P7: P4(1) = P(1, J): U4(1) = U6
1470 U2(I) = U6 + (U7 - U6) * (X2(I) - X6) / (X7 - X6)
1480 P2(I) = P6 + (P7 - P6) * (X2(I) - X6) / (X7 - X6)
1490 A2(I) = FNA(P2(I))
1500 GOSUB 1610
1510 U4(I + 1) = 2 * (A4(I) - A2(I)) / (G0 - 1) + U2(I)
1520 X2(I + 1) = X6 - DT * (U2(I) + U4(I + 1) - A2(I) - A4(I)) / 2
1530 IF ABS(U4(I + 1) - U4(I)) < .1 THEN 1560
1540 X2(I) = X2(I + 1): U4(I) = U4(I + 1)
1550 I = I + 1: GOTO 1470
1560 P(1, J + 1) = P0 * (A4(I) / A0) ^ ((2 * G0) / (G0 - 1))
1570 IF B$ = "n" THEN 1590
1580 PRINT "U(1,J+1)="; U4(I + 1); "P(1,J+1)="; P(1, J + 1)
1590 U(1, J + 1) = U4(I + 1): P(1, J + 1) = P(1, J + 1)
1600 RETURN
1610 Y0 = (2 * A2(I) - U2(I) * (G0 - 1)) / (G0 + 1)
1620 Y1 = Y0 * Y0 + (FNA(PA)) ^ 2 * (G0 - 1) / (G0 + 1)
1630 Y2 = Y1 - 2 * A2(I) ^ 2 / (G0 + 1) - (G0 - 1) ^ 2 * U2(I) ^ 2 / (2 * (G0 + 1))
1640 Y2 = Y2 + 2 * U2(I) * A2(I) * (G0 - 1) / (G0 + 1)
1650 A4(I) = Y0 + SQR(Y2)
1660 P4(I) = P0 * (A4(I) / A0) ^ (2 * G0 / (G0 - 1))
1670 P(1, J) = P4(I): U(1, J) = U4(I)
1680 IF B$ = "n" THEN 1700
1690 PRINT "I="; I, "A4(I)="; A4(I); "U4(I)="; U4(I); "P4(I)="; P4(I)
1700 RETURN
1710 END
1720 I = 1
1730 X5 = X(C - 1): P5 = P(C - 1, J): U5 = U(C - 1, J): X6 = X(C): U6 = U(C, J): P6 = P(C,
     J)
1740 X7 = X(C + 1): U7 = U(C + 1, J): P7 = P(C + 1, J)
1750 X1(1) = X5: X2(1) = X7: P1(1) = P5: P2(1) = P7: P4(1) = P6: U4(1) = U6
1760 U1 = U6 + (U5 - U6) * (X1(I) - X6) / (X5 - X6)
1770 U1(I) = U1
1780 P1 = P6 + (P5 - P6) * (X1(I) - X6) / (X5 - X6)
1790 P1(I) = P1
1800 U2 = U6 + (U7 - U6) * (X2(I) - X6) / (X7 - X6)
1810 A1(I) = FNA(P1(I))
1820 U2(I) = U2
1830 P2 = P6 + (P7 - P6) * (X2(I) - X6) / (X7 - X6)
1840 P2(I) = P2
1850 A2(I) = FNA(P2(I))
1860 A4(I) = FNA(P4(I))
1870 X1(I + 1) = X6 - DT * (U1(I) + A1(I) + U4(I) + A4(I)) / 2
1880 X2(I + 1) = X6 - DT * (U2(I) + U4(I) - A2(I) - A4(I)) / 2
1890 U4(I + 1) = (FNA(P1(I)) - FNA(P2(I))) / (G0 - 1) + (U1(I) + U2(I)) / 2
1900 A4(I + 1) = FNA(P2(I)) + (G0 - 1) * (U4(I + 1) - U2(I)) / 2
1910 P4(I + 1) = P0 * (A4(I + 1) / A0) ^ ((2 * G0) / (G0 - 1))
1920 IF ABS(U4(I + 1) - U4(I)) < .1 THEN 1960
1930 X1(I) = X1(I + 1): X2(I) = X2(I + 1): P4(I) = P4(I + 1): U4(I) = U4(I + 1)
1940 I = I + 1
```

```
1950  GOTO 1760
1960  U(C, J + 1) = U4(I + 1): P(C, J + 1) = P4(I + 1)
1970  RETURN
1980  X5 = X(NX - 1): P5 = P(NX - 1, J): U5 = U(NX - 1, J): X6 = X(NX): P6 = P(NX, J):
      U6 = U(NX, J): I = 1
1990  X1(1) = X(NX - 1): P1(1) = P(NX - 1, J): P4(1) = P(NX, J): U4(1) = 0
2000  U1(I) = U5 * (X1(I) - X6) / (X5 - X6)
2010  P1(I) = P6 + (P5 - P6) * (X1(I) - X6) / (X5 - X6)
2020  A1(I) = FNA(P1(I))
2030  A4(I) = FNA(P4(I))
2040  X1(I + 1) = X6 - DT * (U1(I) + A1(I) + U4(I) + A4(I)) / 2
2050  A4(I + 1) = A1(I) + U1(I) * ((G0 - 1) / 2)
2060  P4(I + 1) = P0 * (A4(I + 1) / A0) ^ ((2 * G0) / (G0 - 1))
2070  IF ABS(P4(I + 1) - P4(I)) < 50 THEN 2110
2080  X1(I) = X1(I + 1): P4(I) = P4(I + 1)
2090  I = I + 1
2100  GOTO 2000
2110  IF R$ = "y" AND P4(I + 1) < PR THEN 2720
2120  IF B$ = "n" THEN 2140
2130  PRINT "at closed end P4="; P4(I + 1); "  at time (sec)="; J * DT
2140  U(NX, J + 1) = 0: P(NX, J + 1) = P4(I + 1)
2160  GOSUB 3690
2170  PRINT USING "#####.####"; DIST(J + 1); J; DT * J; C - 1; C; Y(J); Y(J + 1); UAV(J)
2180  IF DIST(J + 1) > 0 THEN 2210
2190  IF COUNTER<>0 THEN 2250
2200  IJ = J: PRINT "IJ="; IJ: GOTO 2220
2210  IF DIST(J + 1) > 0 THEN 2310
2220  PRINT "Interphase Line has Reached Left End"
2230  DIST(J) = 0: LPRINT "IJ="IJ
2240  COUNTER=1
2250  Q$ = "y"
2260  IF J > NJ THEN 1090
2270  IF B$ = "n" THEN 2310
2280  PRINT
2290  PRINT "New time line started for t (ms)="; J * DT
2300  IF COUNTER<>0 THEN 2330
2310  DMASS=-FNR(P(1,J))*U(1,J)*DT
2320  MASS(J+1)=MASS(J)+DMASS
2330  IF J >= LEC THEN 3620
2340  IF J=REA THEN  3640
2350  IF J >= LEO THEN 3630
2360  IF COUNTER<>0 THEN 2390
2370  IF P(1,J)<PA THEN 3620
2380  IF COUNTER=0 THEN 3660
2390  IF J>IJ THEN 3620
2400  GOTO 3630
2410  LPRINT Q$
2420  RETURN
2430  END
2440  LINE INPUT "FILENAME ?"; FILE$
2450  PRINT "Data Stored in File"; FILE$; ".PRN"
2460  LPRINT "Data Stored in File"; FILE$; ".PRN"
2470  OPEN FILE$ + ".PRN" FOR OUTPUT AS #2
2480  FOR C = 1 TO NX
```

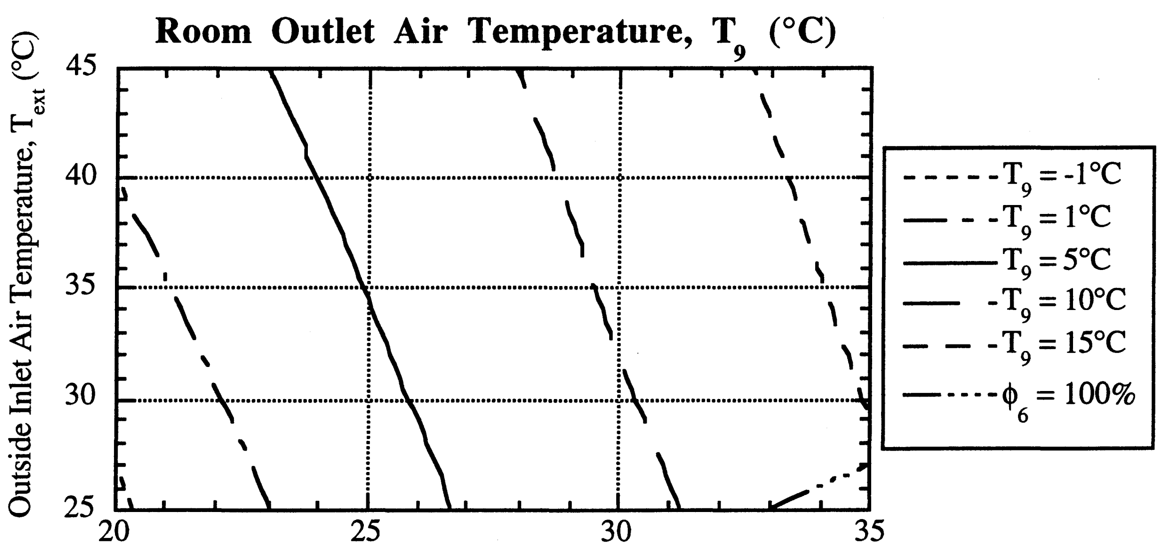
```
2490  FOR J = 1 TO NJ
2500  PRINT #2, (J - 1) * DT, P(C, J), U(C, J)
2510  NEXT J
2520  NEXT C
2530  FOR J = 1 TO NJ
2540  PRINT #2, (J - 1) * DT, DIST(J)
2550  NEXT J
2560  CLOSE #2
2570  GOTO 1100
2580  U = U(1, J)
2590  AV = A0 + (G0 - 1) * U / 2
2600  MI = U / AV
2610  PI = (AV / A0) ^ (2 * G0 / (G0 - 1))
2620  P0I = (1 + (G0 - 1) * MI * MI / 2) ^ (G0 / (G0 - 1)) * PI
2630  PL = P0I * P0 / PA
2640  IF PL > 1 THEN 2660
2650  PL = 1 / PL
2660  MA = SQR(2 * (PL ^ ((G0 - 1) / G0) - 1) / (G0 - 1))
2670  AR = FNU(MI) / FNU(MA)
2680  PRINT C, MI, MA, -1 / AR
2690  P(1, J) = PA * ((1 + (G0 - 1) * MA * MA / 2) / (1 + (G0 - 1) * MI * MI / 2)) ^ (G0 / (G0 -
      1))
2700  RETURN
2710  END
2720  X5 = X(NX - 1): P5 = P(NX - 1, J): U5 = U(NX - 1, J): X6 = X(NX): P6 = P(NX, J):
      U6 = U(NX, J): I = 1
2730  IF RO > 0 THEN 2780
2740  PRINT "REEDVALVE OPENS, J="; J: J0 = J: DIST(J0) = L
2750  UAV(J0) = U(NX, J0): PRINT "UAV(J0)="; UAV(J0)
2760  RO = 1
2770  IF B$ <> "y" THEN 2800
2780  IF B$ <> "y" THEN 2800
2790  PRINT "REEDVALVE OPEN J="; J
2800  X1(1) = X5: P1(1) = P5: P4(1) = P6: U4(1) = U6
2810  U1(I) = U6 + (U5 - U6) * (X1(I) - X6) / (X5 - X6)
2820  P1(I) = P6 + (P5 - P6) * (X1(I) - X6) / (X5 - X6)
2830  A1(I) = FNA(P1(I))
2840  GOSUB 2920
2850  U4(I + 1) = -2 * (A4(I) - A1(I)) / (G0 - 1) + U1(I)
2860  X1(I + 1) = X6 - DT * (U1(I) + U4(I + 1) + A1(I) + A4(I)) / 2
2870  IF ABS(U4(I + 1) - U4(I)) < .1 THEN 2900
2880  X1(I) = X1(I + 1): U4(I) = U4(I + 1)
2890  I = I + 1: GOTO 2810
2900  U(NX, J + 1) = U4(I): P(NX, J + 1) = P4(I)
2910  GOTO 2160
2920  Y0 = (2 * A1(I) + U1(I) * (G0 - 1)) / (G0 + 1)
2930  Y1 = Y0 * Y0 + (FNA(PR)) ^ 2 * (G0 - 1) / (G0 + 1)
2940  Y2 = Y1 - (G0 + 1) * Y0 * Y0 / 2
2950  A4(I) = Y0 + SQR(Y2)
2960  P4(I) = P0 * (A4(I) / A0) ^ ((2 * G0) / (G0 - 1))
2970  IF B$ = "n" THEN 2990
2980  PRINT "I="; I; " A4(I)="; A4(I); " U4(I)="; U4(I); " P4(I)="; P4(I)
2990  RETURN
3000  END
```

```
3010 PRINT "enter station # for PRINTED record,(pressure and velocity vs time)";
3020 INPUT C0
3030 PRINT "Pressure record at station "; C0; "  where X="; X(C0)
3040 PRINT TAB(5); "Time, (sec)"; TAB(25); "Pressure, (Pa)"; TAB(45); "Velocity, (m/sec)"
3050 FOR C = 1 TO NX
3060 FOR J = 1 TO NJ
3070 IF C <> C0 THEN 3090
3080 PRINT TAB(5); (J - 1) * DT; TAB(25); P(C, J); TAB(45); U(C, J)
3090 NEXT J
3100 NEXT C
3110 PRINT "is another station record to be printed (y/n)?"
3120 INPUT T$
3130 IF T$ = "y" THEN 3010
3140 GOTO 4000
3150 I = 1
3160 PRINT "J="; J
3170 X7 = X(2): P7 = P(2, J): U6 = 0: X6 = 0: U7 = U(2, J): P6 = P(1, J)
3180 PRINT X7, X6, P7, P6
3190 I = 1
3200 X2(1) = X7: P2(1) = P7: P4(1) = P(1, J): U4(1) = 0
3210 U2(I) = U7 * X2(I) / X7
3220 P2(I) = P6 + (P7 - P6) * X2(I) / X7
3230 A2(I) = FNA(P2(I))
3240 A4(I) = FNA(P4(I))
3250 X2(I + 1) = -DT * (U2(I) - A4(I) - A2(I)) / 2
3260 A4(I + 1) = A2(I) - U2(I) * ((G0 - 1) / 2)
3270 P4(I + 1) = P0 * (A4(I + 1) / A0) ^ ((2 * G0) / (G0 - 1))
3280 PRINT P4(I + 1), P4(I)
3290 IF ABS(P4(I + 1) - P4(I)) < 50 THEN 3320
3300 X2(I) = X2(I + 1): P4(I) = P4(I + 1): I = I + 1
3310 GOTO 3210
3320 P(1, J + 1) = P4(I + 1)
3330 IF B$ = "n" THEN 3350
3340 PRINT "P(1,J+1)="; P4(I + 1): P(1, J + 1) = P4(I + 1)
3350 RETURN
3360 IF B$ = "n" THEN 3380
3370 PRINT "Critical flow, left end at station #1, x=0"
3380 X7 = X(2): P7 = P(2, J): U6 = U(1, J): X6 = 0: U7 = U(2, J): P6 = P(1, J)
3390 I = 1
3400 X2(1) = X7: P2(1) = P7: P4(1) = P(1, J): U4(1) = U6
3410 U2(I) = U6 + (U7 - U6) * (X2(I) - X6) / (X7 - X6)
3420 P2(I) = P6 + (P7 - P6) * (X2(I) - X6) / (X7 - X6)
3430 A2(I) = FNA(P2(I))
3440 A4(I) = FNA(P4(I))
3450 X2(I + 1) = X6 - DT * (U2(I) + U4(I) - A2(I) - A4(I)) / 2
3460 U4(I + 1) = -2 * FNA(P2(I)) / (G0 + 1) + U2(I) * (G0 - 1) / (G0 + 1)
3470 A4(I + 1) = -U4(I + 1)
3480 P4(I + 1) = P0 * (A4(I + 1) / A0) ^ ((2 * G0) / (G0 - 1))
3490 IF ABS(U4(I + 1) - U4(I)) < .001 THEN 3590
3500 X2(I) = X2(I + 1): U4(I) = U4(I + 1): P(1, J) = P4(I + 1)
3510 I = I + 1
3520 GOTO 3410
3530 IF B$ = "n" THEN 3550
3540 PRINT "U(1,J+1)="; U4(I + 1); "P(1,J+1)="; P4(1)
```

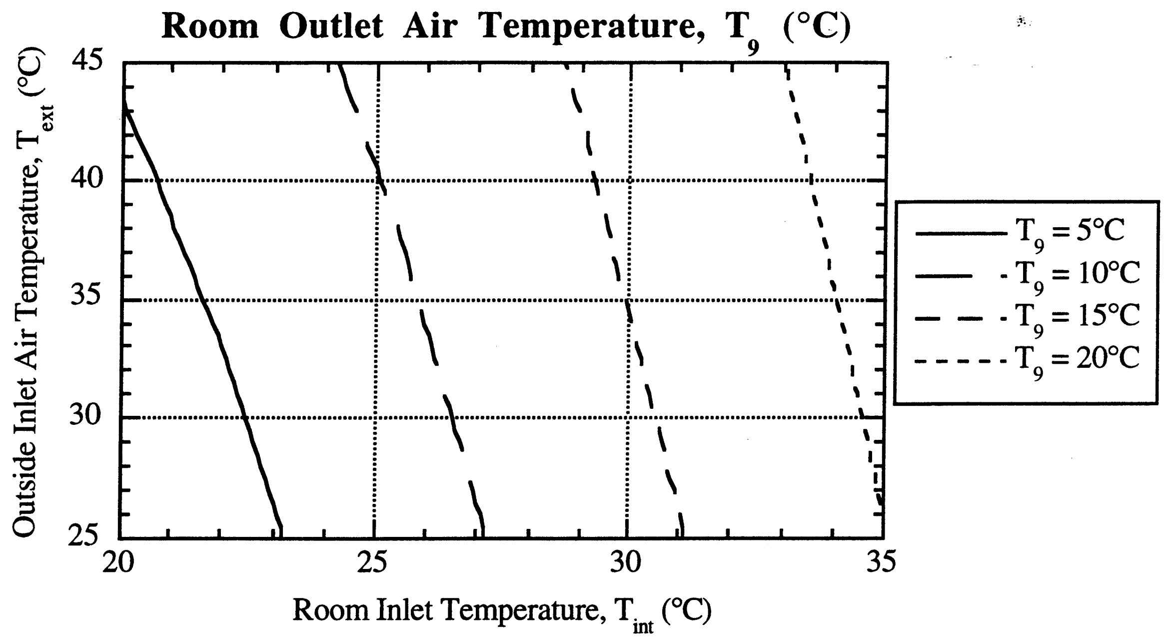
```
3550 U(1, J + 1) = U4(I + 1): P(1, J + 1) = P4(I + 1)
3560 IF U4(I + 1) > 0 THEN 1430
3570 IF B$ = "n" THEN 3590
3580 PRINT "Outflow"
3590 U(1, J + 1) = U4(I + 1): P(1, J + 1) = P4(I + 1)
3600 PRINT J, U(1, J + 1), P(1, J + 1)
3610 RETURN
3620 Q$ = "y": PRINT "Left End now CLOSED"
3621 IF FL1<>0 THEN 3624
3622 LPRINT "Left End Closed at J="J
3623 FL1=1
3624 GOTO 2420
3630 Q$ = "n": PRINT "Left End now Opened":PA=P0
3631 IF FL2<>0 THEN 3635
3632 LPRINT "Left End Opens at J="J
3633 FL2=1
3635 GOTO 2420
3640 IF J=REA THEN 3680
3650 PR=P0: GOTO 2350
3660 Q$="n":PRINT "Left end is now OPENED"
3661 IF FL3<>0 THEN 3665
3662 LPRINT "Left End Opens at J="J
3663 FL3=1
3665 GOTO 2420
3670 END
3680 DIST(REA)=L:GOSUB 3690:GOTO 3650
3690 IF DIST(J) <= 0 THEN 2190
3700 FOR C = 2 TO NX
3710 IF DIST(J) > (C - 2) * L / (NX - 1) AND DIST(J) <= (C - 1) * L / (NX - 1) THEN 3730
3720 NEXT
3730 UAV(J) = U(C - 1, J) + ((U(C, J) - U(C - 1, J)) * (Y(J) - (C - 1) * L / (NX - 1)) / (L /
     (NX - 1)))
3740 Y(J + 1) = Y(J) + UAV(J) * DT: DIST(J + 1) = L + Y(J + 1)
3750 IF SGN(DIST(J+1))/SGN(DIST(J))<0 AND J>LEO THEN 3770
3760 RETURN
3770 LEC=J:GOTO 3760
4000 LPRINT "Initial Distribution for the NEXT CYCLE"
4010 FOR C=1 TO NX
4020 LPRINT "U("C",1)=";U(C,LEC+15),"P("C",1)="P(C,LEC+15)
4025 U(C,1)=U(C,LEC+15):P(C,1)=P(C,LEC+15)
4030 NEXT
4035 GOTO 685
4040 END
```



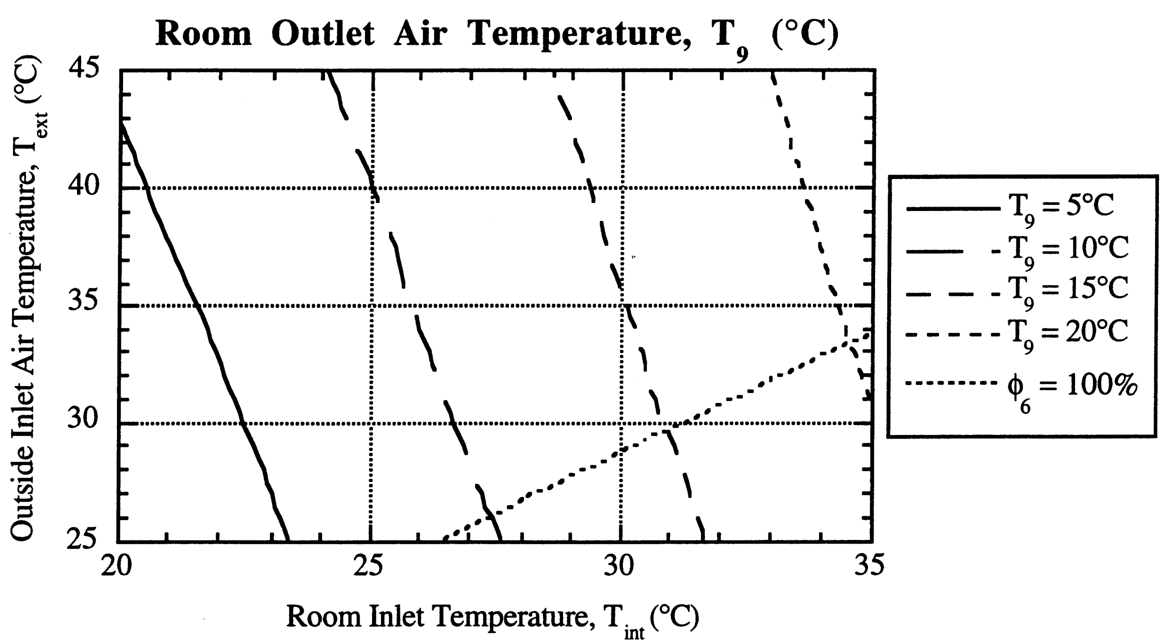
Room Inlet Temperature, T_{int} (°C)
Relative Humidity = 55%
Heat Exchanger Effectiveness = 70%



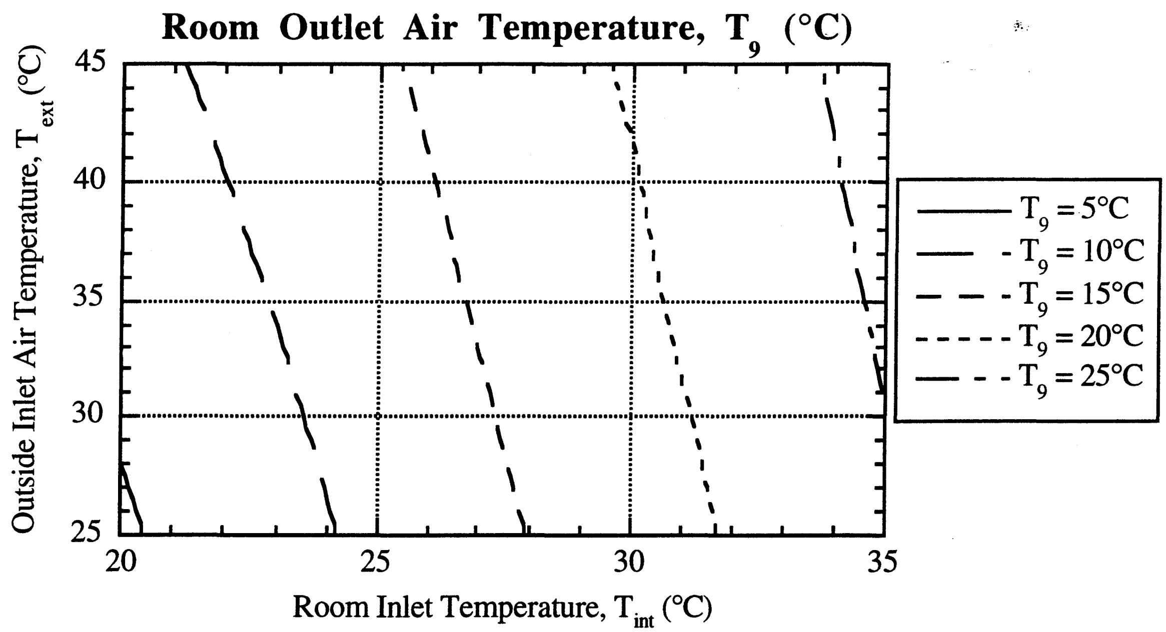
Room Inlet Temperature, T_{int} (°C)
Relative Humidity = 55%
Heat Exchanger Effectiveness = 90%



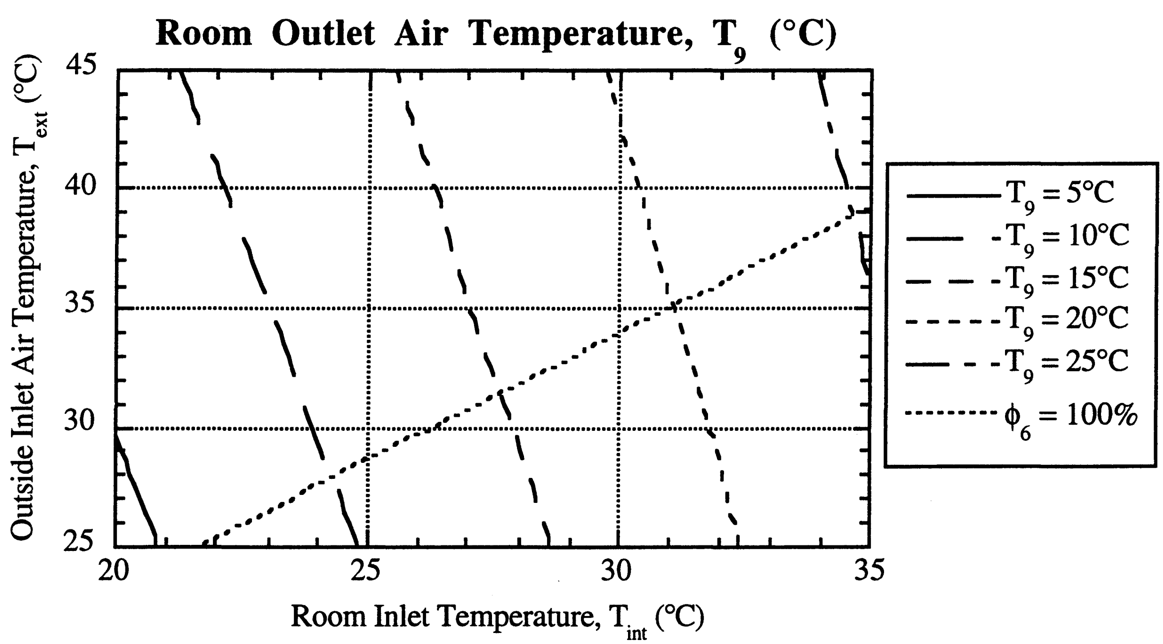
Relative Humidity = 75%
Heat Exchanger Effectiveness = 70%



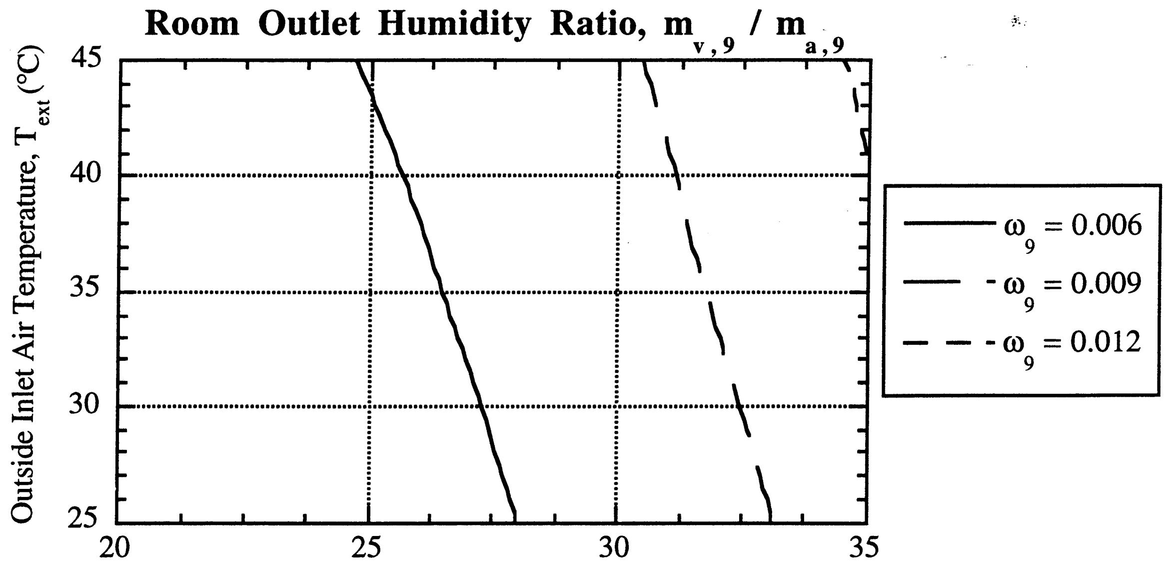
Relative Humidity = 75%
Heat Exchanger Effectiveness = 90%



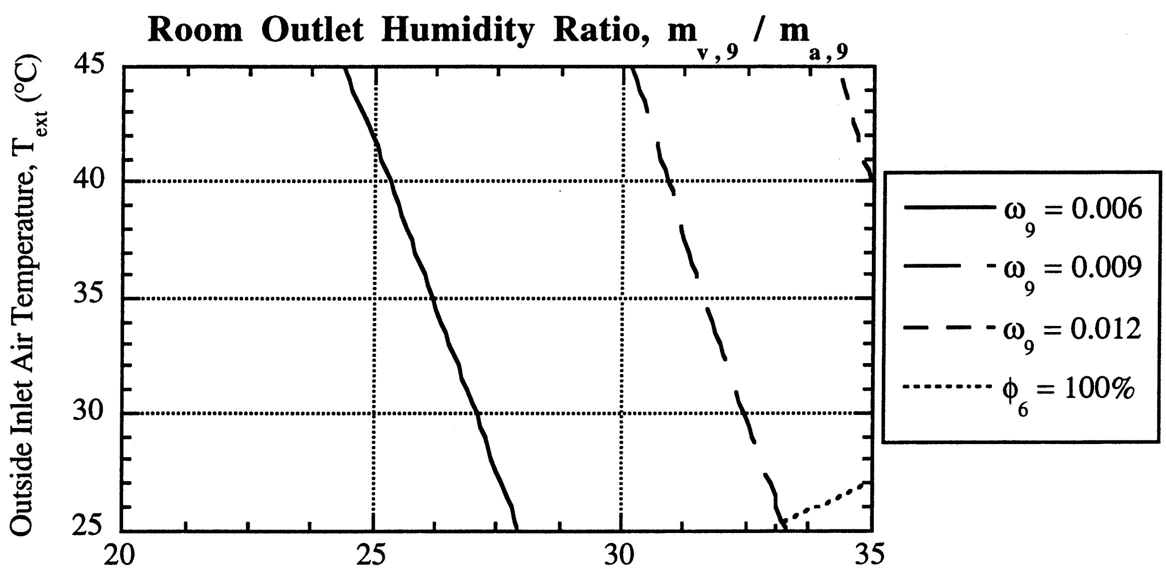
Relative Humidity = 95%
Heat Exchanger Effectiveness = 70%



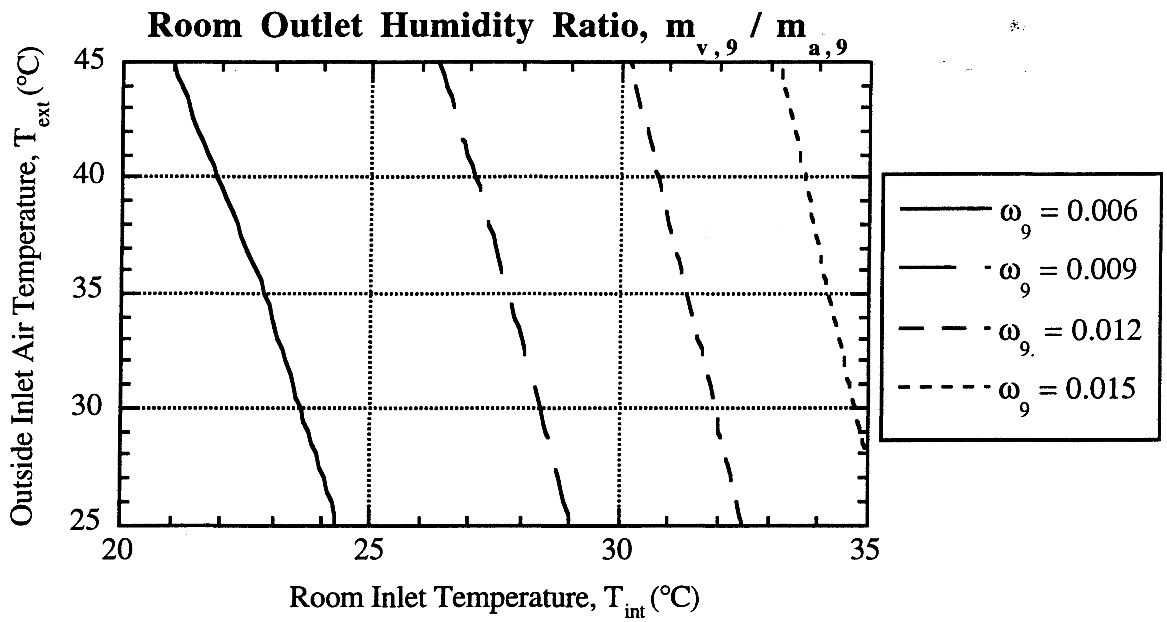
Relative Humidity = 95%
Heat Exchanger Effectiveness = 90%



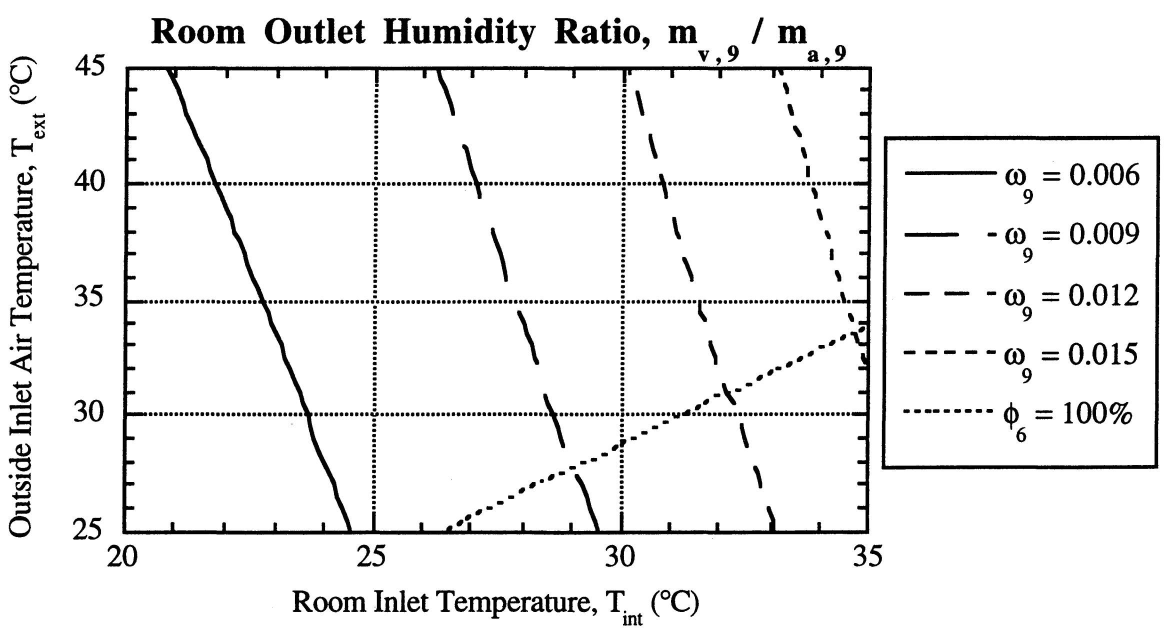
Room Inlet Temperature, T_{int} (°C)
Relative Humidity = 55%
Heat Exchanger Effectiveness = 70%



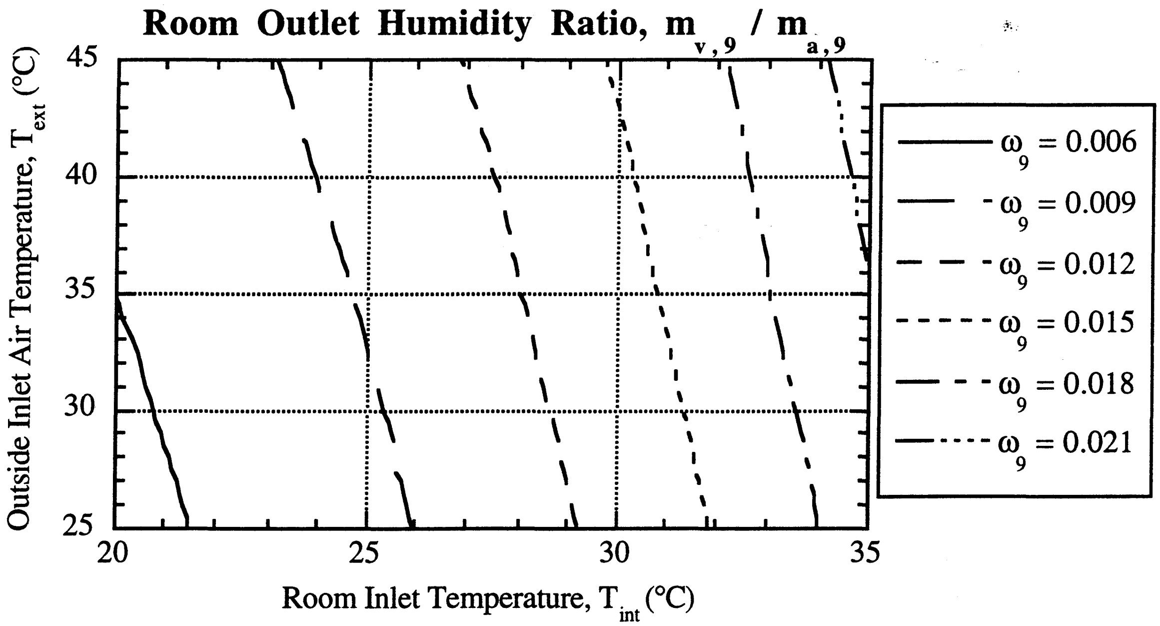
Room Inlet Temperature, T_{int} (°C)
Relative Humidity = 55%
Heat Exchanger Effectiveness = 90%



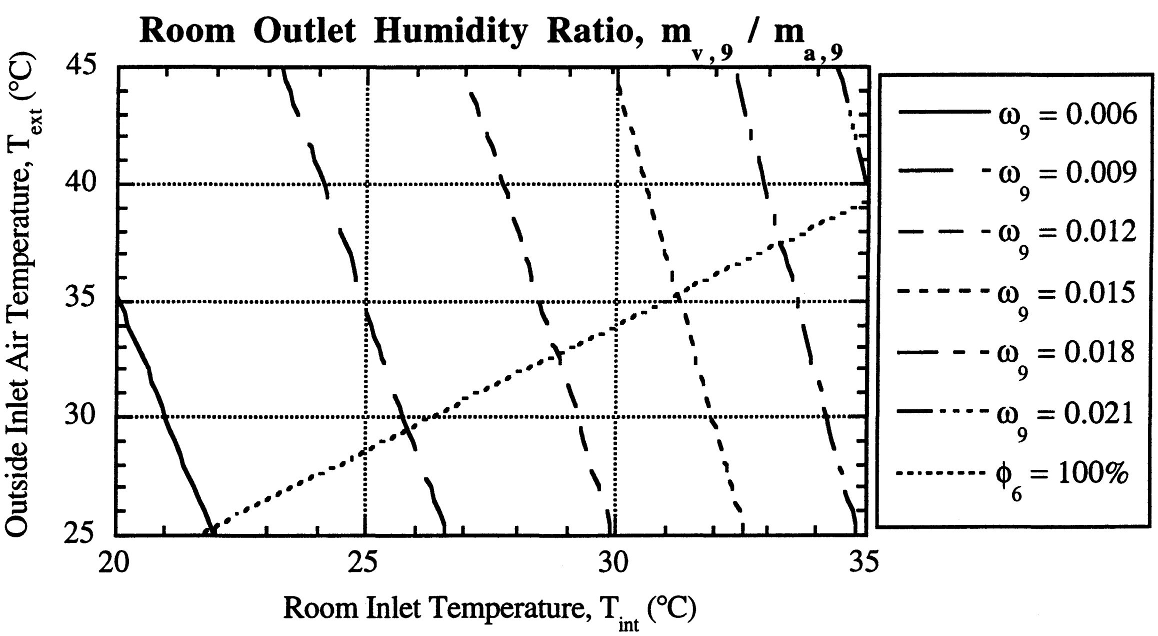
Relative Humidity = 75%
Heat Exchanger Effectiveness = 70%



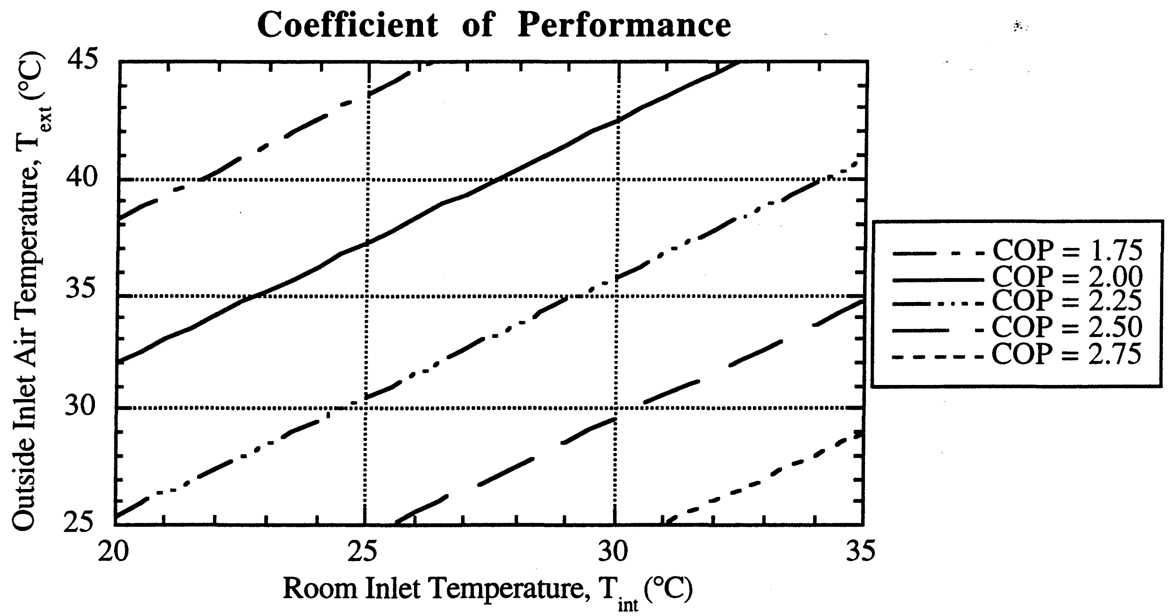
Relative Humidity = 75%
Heat Exchanger Effectiveness = 90%



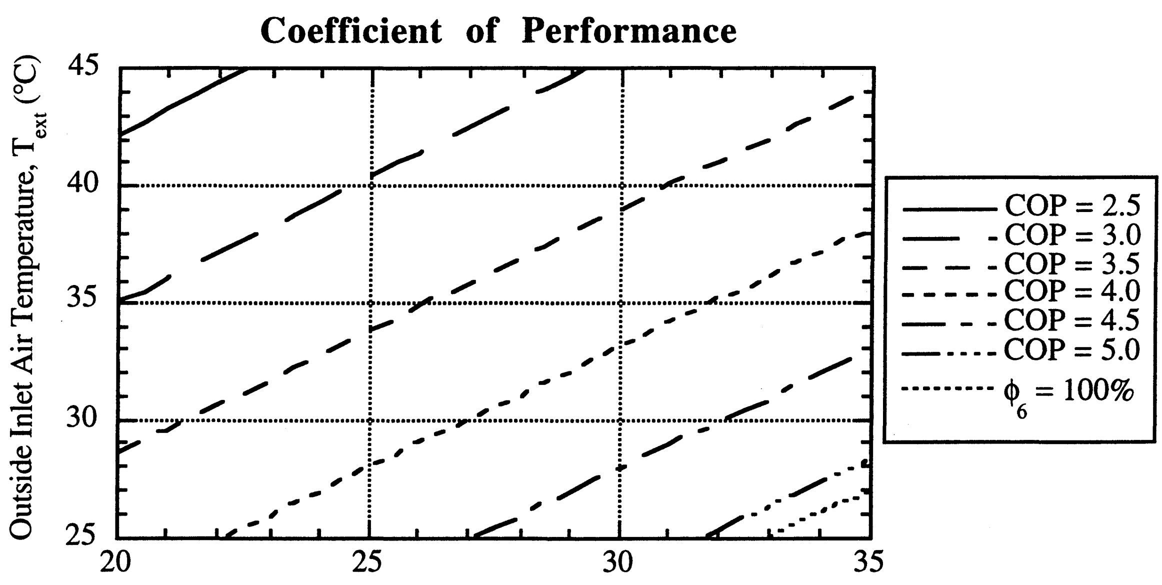
Relative Humidity = 95%
Heat Exchanger Effectiveness = 70%



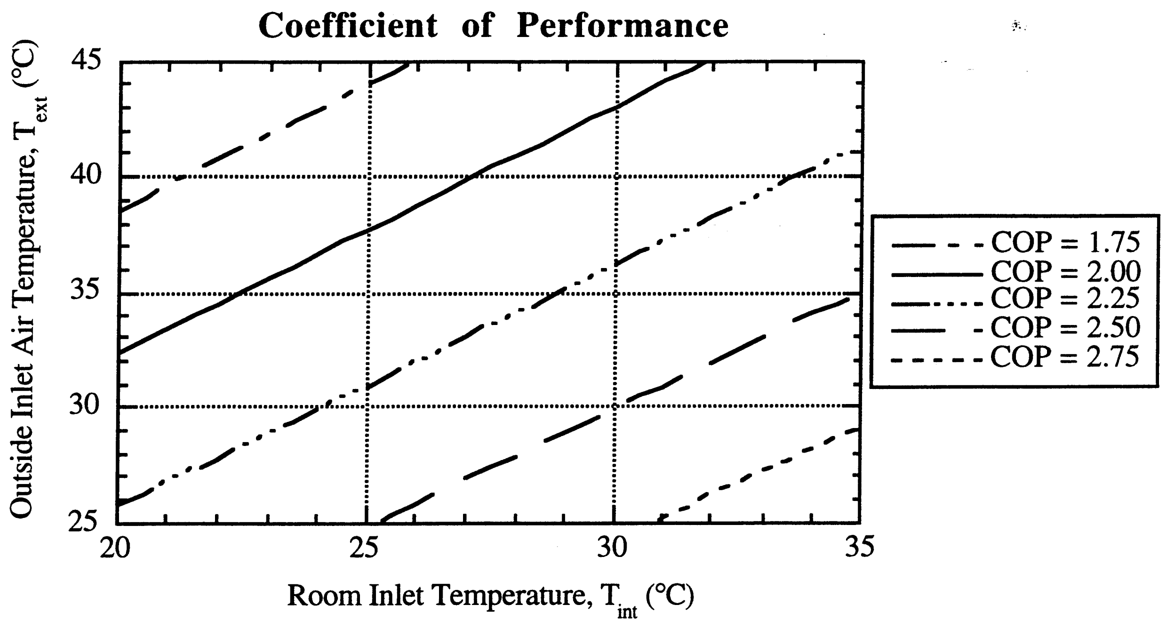
Relative Humidity = 95%
Heat Exchanger Effectiveness = 90%



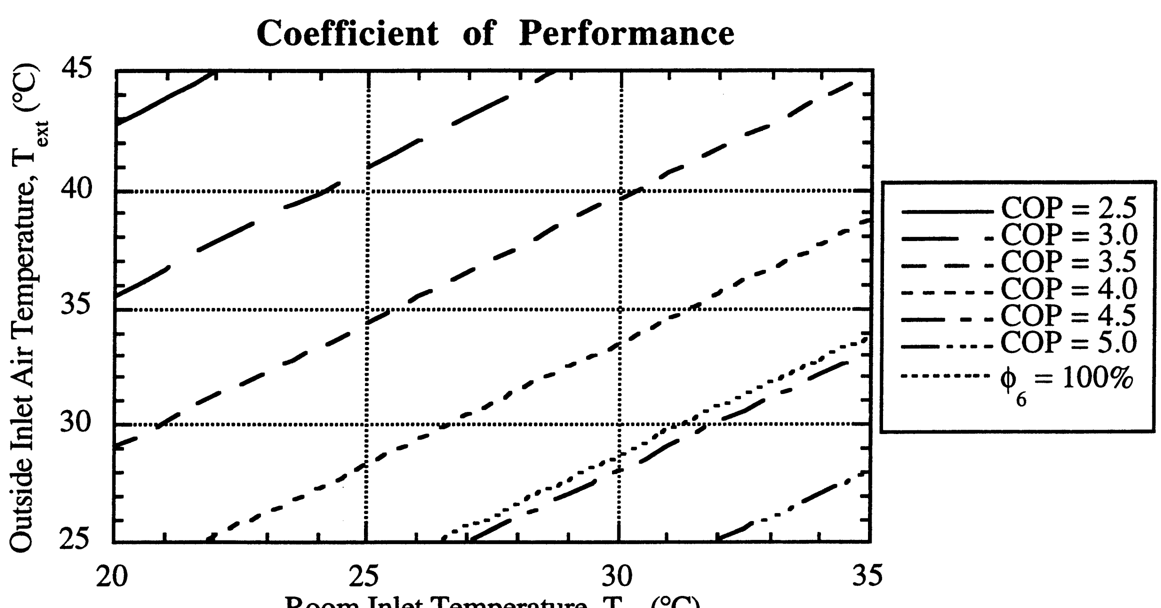
Relative Humidity = 55%
Heat Exchanger Effectiveness = 70%



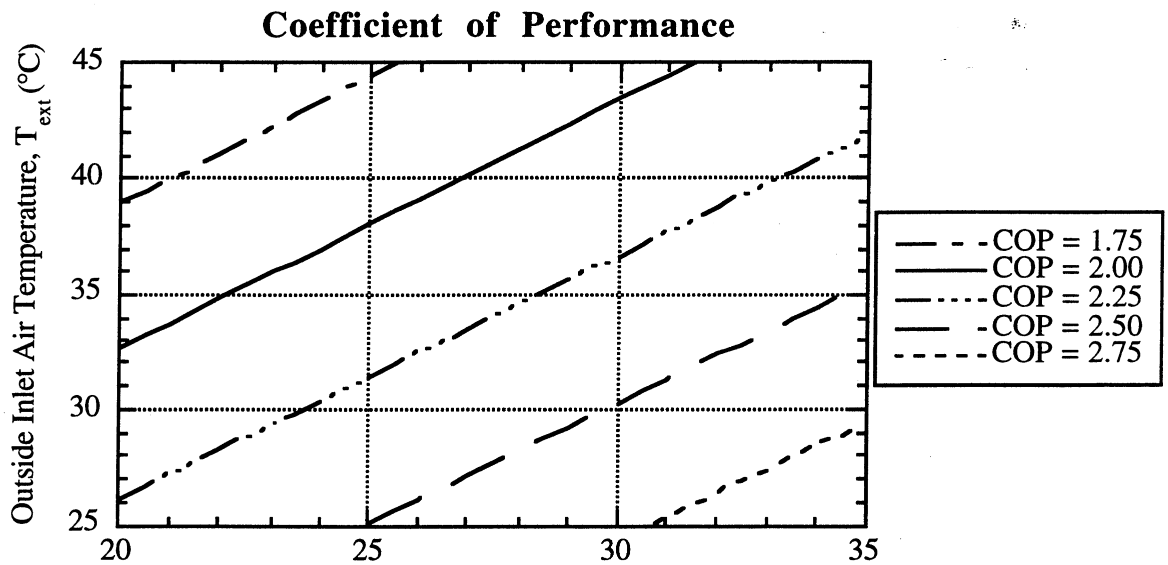
Relative Humidity = 55%
Heat Exchanger Effectiveness = 90%



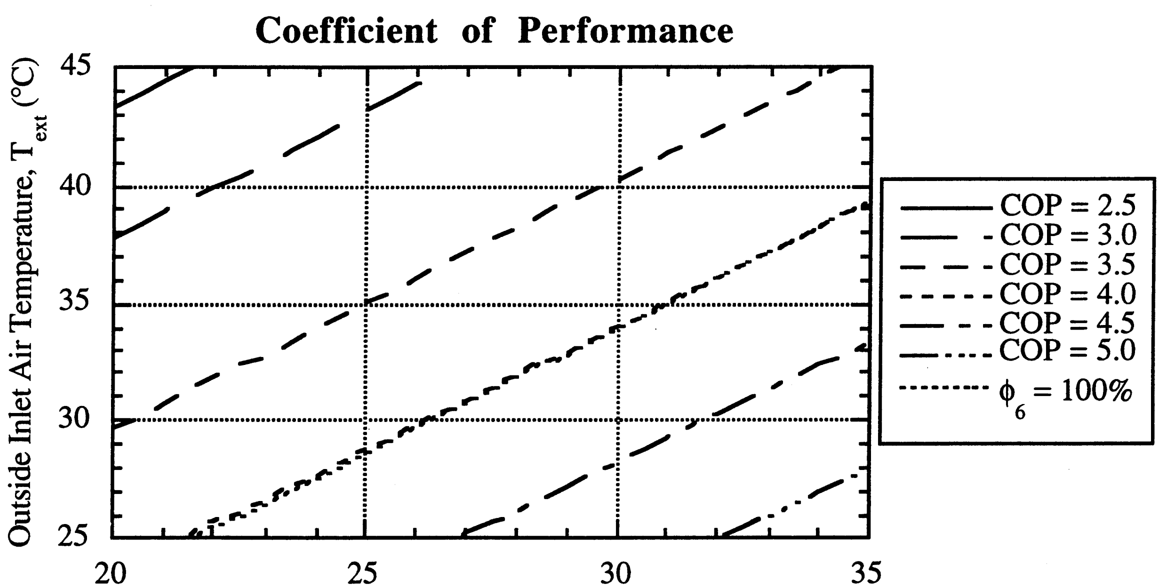
Relative Humidity = 75%
Heat Exchanger Effectiveness = 70%



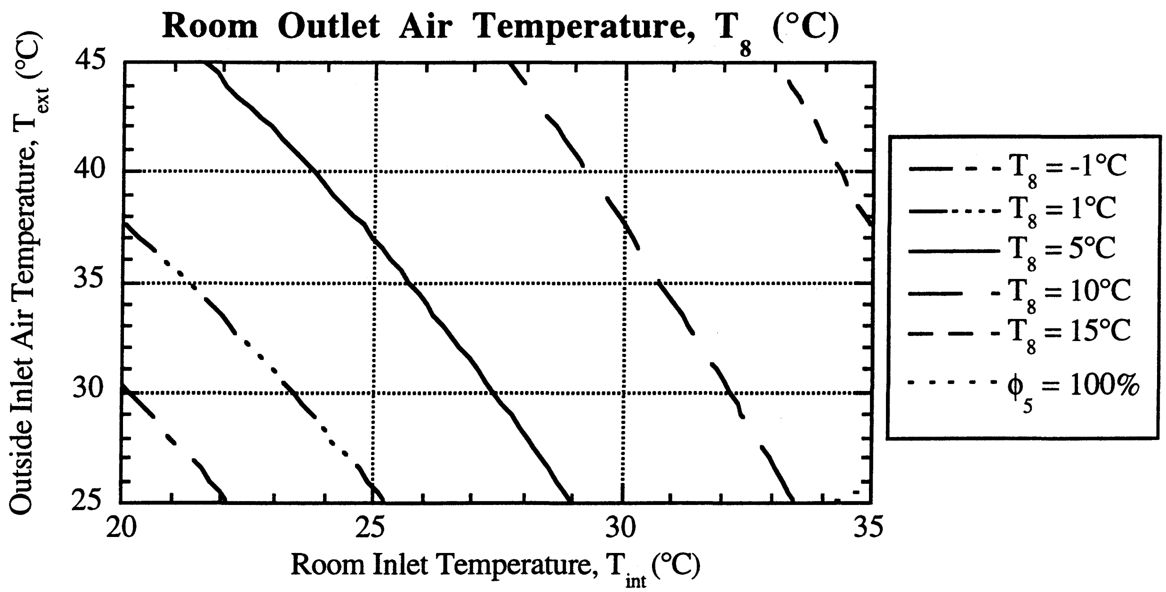
Relative Humidity = 75%
Heat Exchanger Effectiveness = 90%



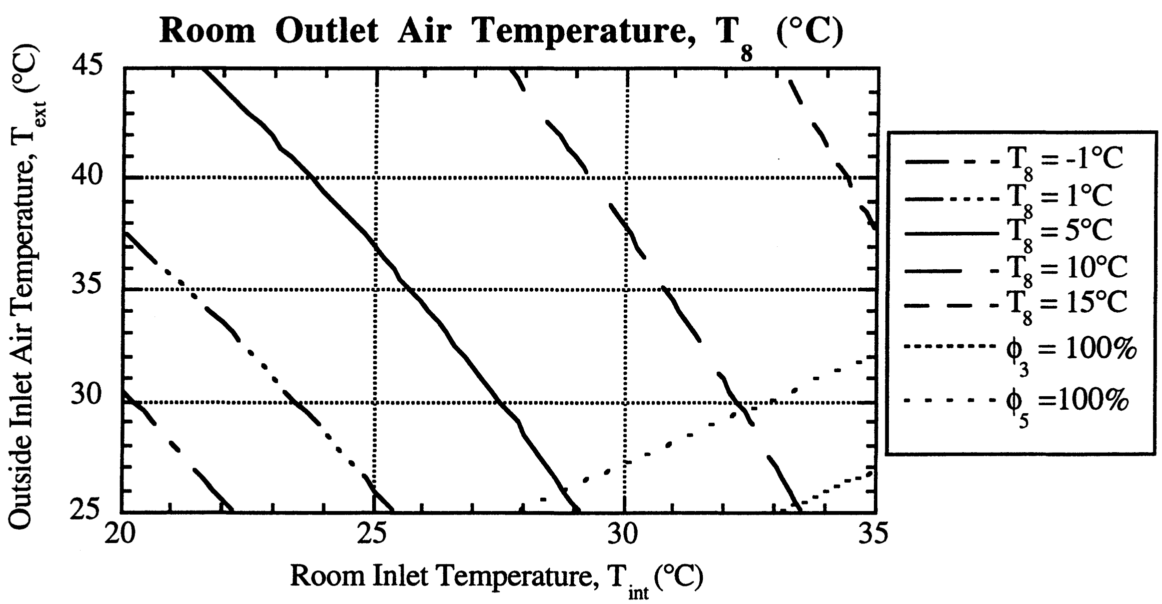
Room Inlet Temperature, T_{int} (°C)
Relative Humidity = 95%
Heat Exchanger Effectiveness = 70%



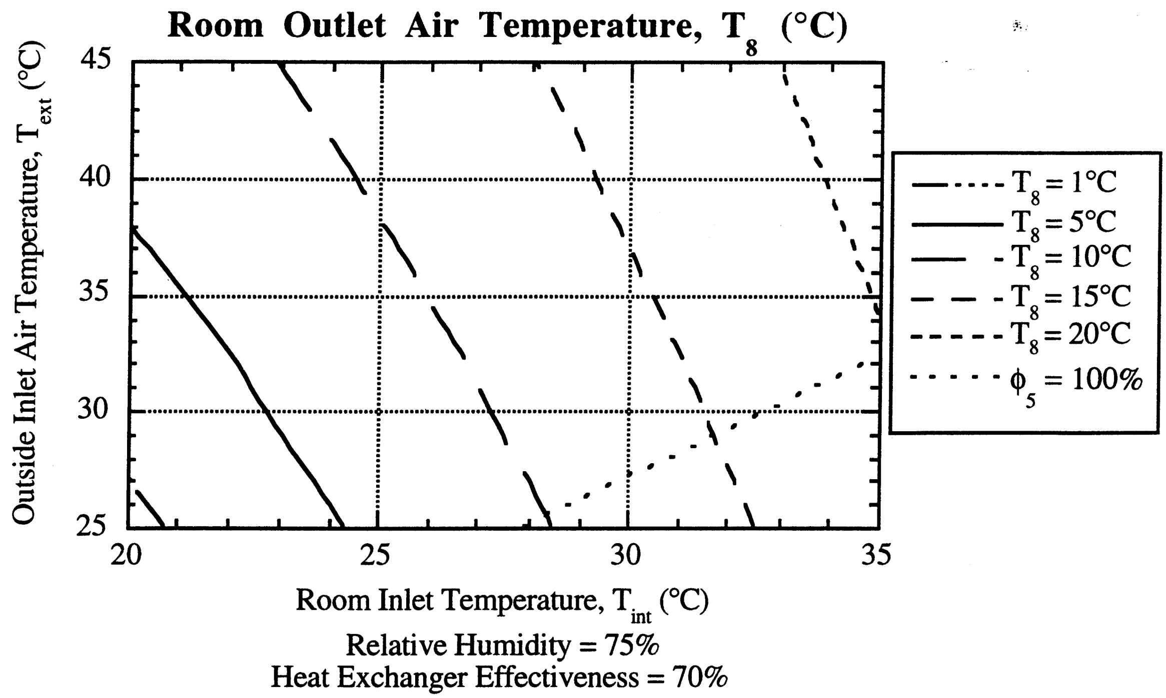
Room Inlet Temperature, T_{int} (°C)
Relative Humidity = 95%
Heat Exchanger Effectiveness = 90%

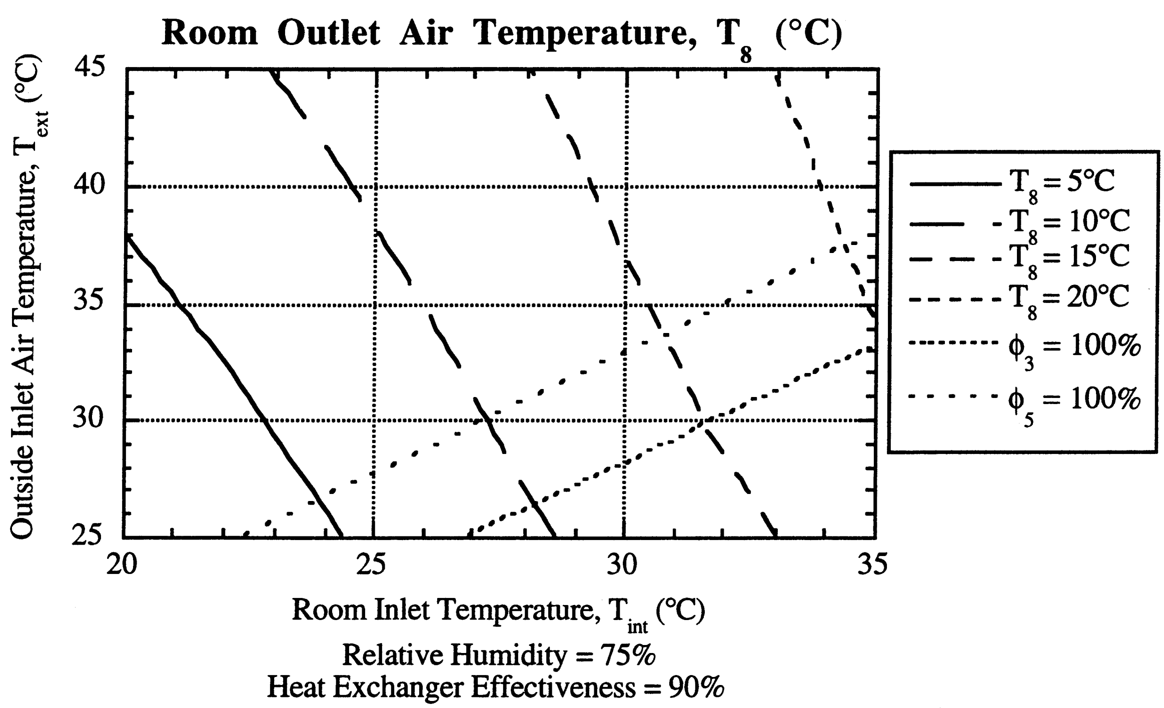


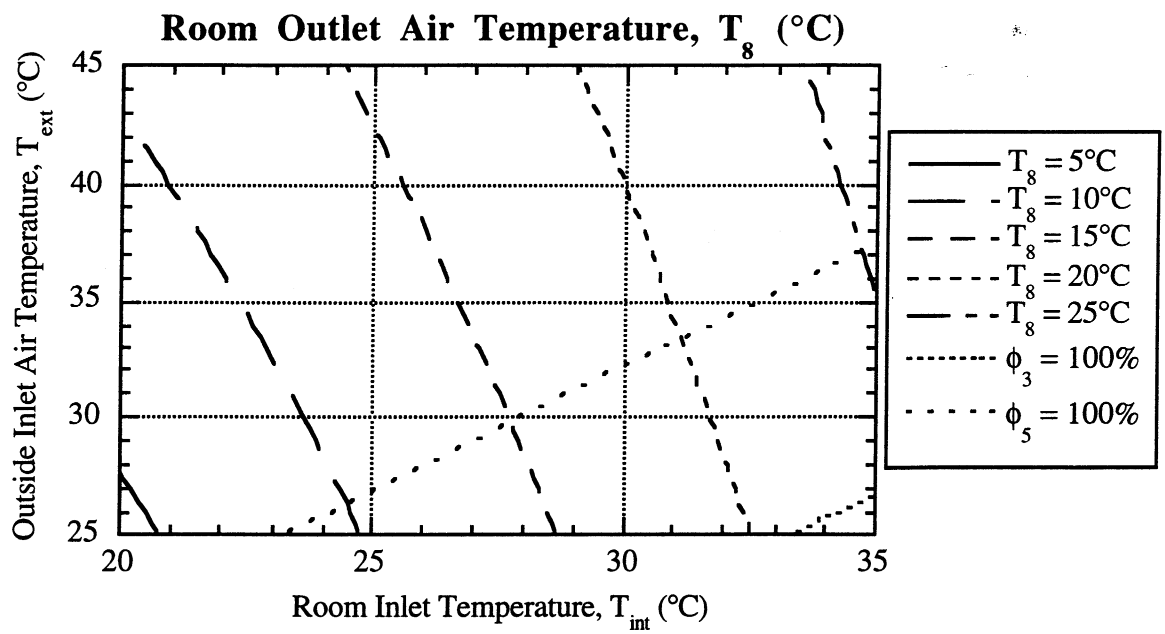
Relative Humidity = 55%
Heat Exchanger Effectiveness = 70%



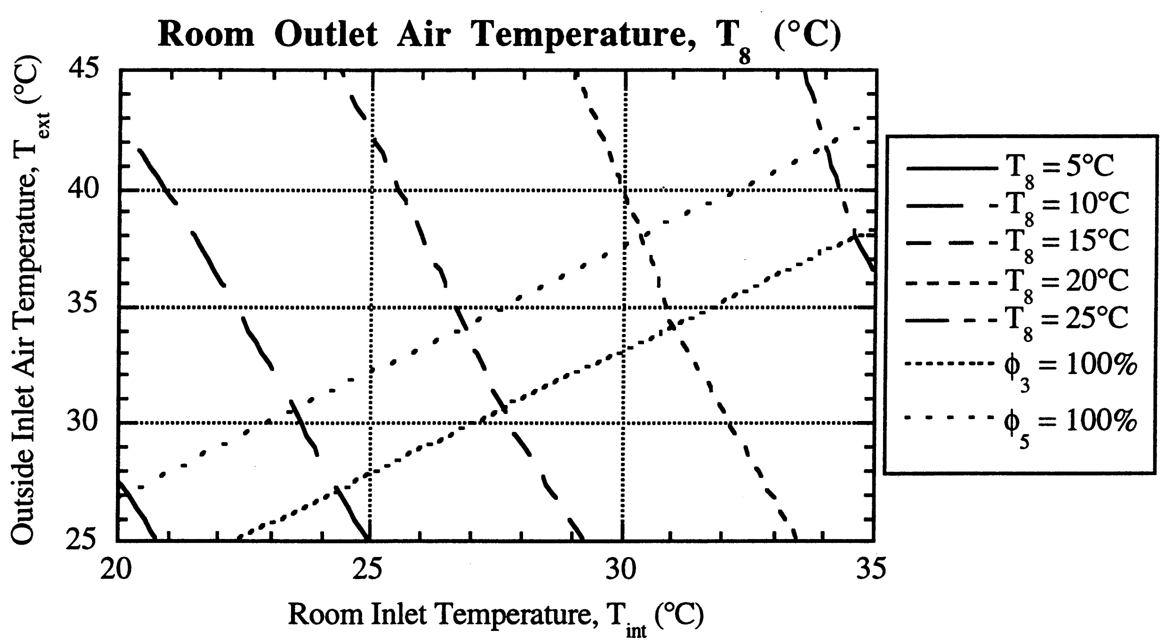
Relative Humidity = 55%
Heat Exchanger Effectiveness = 90%


Room Outlet Air Temperature, T_8 (°C)
Outside Inlet Air Temperature, T_{ext} (°C)
45
40
35
30
25
20
25
30
35
T_8 = 1°C
T_8 = 5°C
T_8 = 10°C
T_8 = 15°C
T_8 = 20°C
ϕ_5 = 100%
Room Inlet Temperature, T_{int} (°C)
Relative Humidity = 75%
Heat Exchanger Effectiveness = 70%

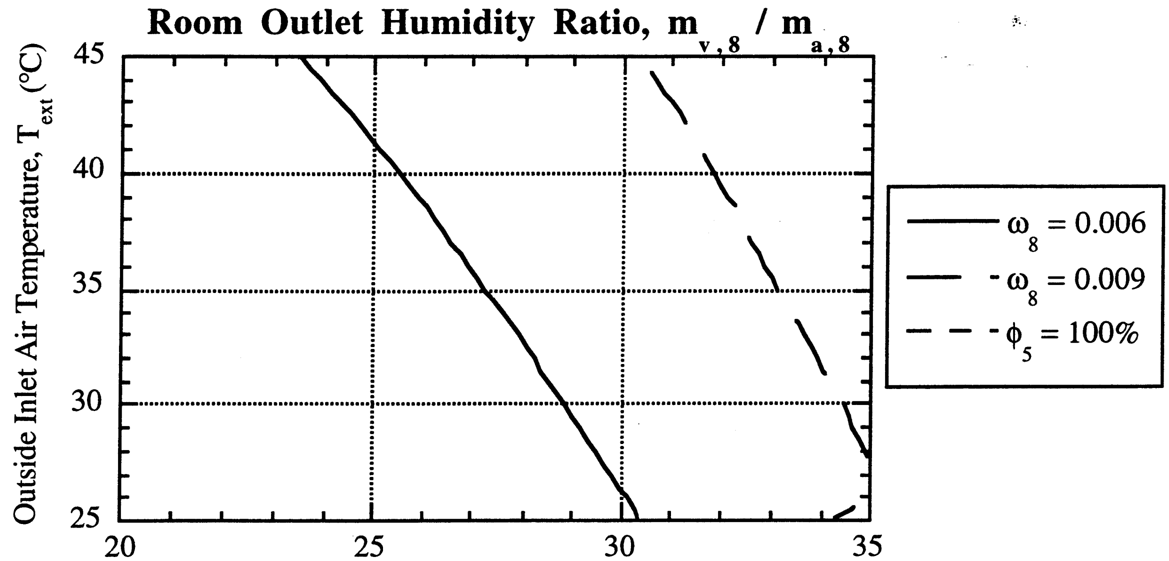

Room Outlet Air Temperature, T_8 (°C)
Outside Inlet Air Temperature, T_{ext} (°C)
45
40
35
30
25
20
25
30
35
T_8 = 5°C
T_8 = 10°C
T_8 = 15°C
T_8 = 20°C
ϕ_3 = 100%
ϕ_5 = 100%
Room Inlet Temperature, T_{int} (°C)
Relative Humidity = 75%
Heat Exchanger Effectiveness = 90%



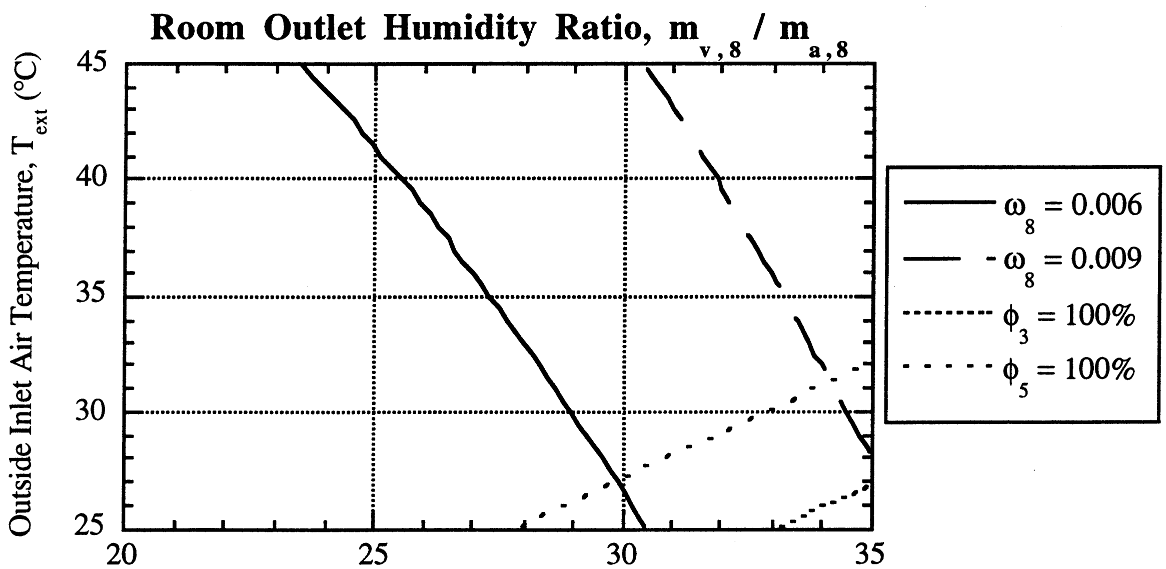
Relative Humidity = 95%
Heat Exchanger Effectiveness = 70%



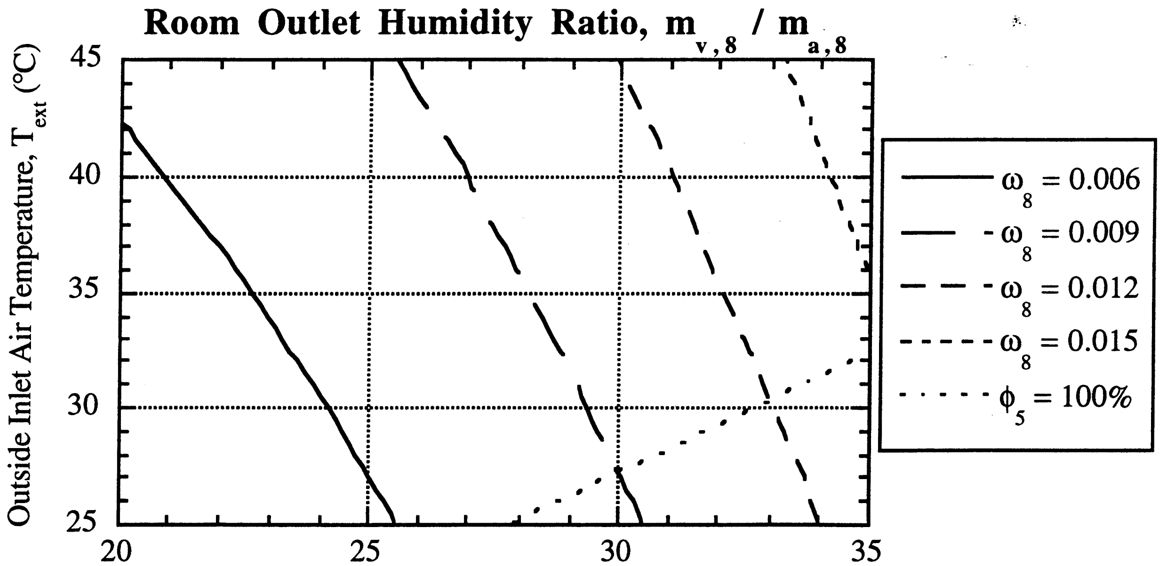
Relative Humidity = 95%
Heat Exchanger Effectiveness = 90%



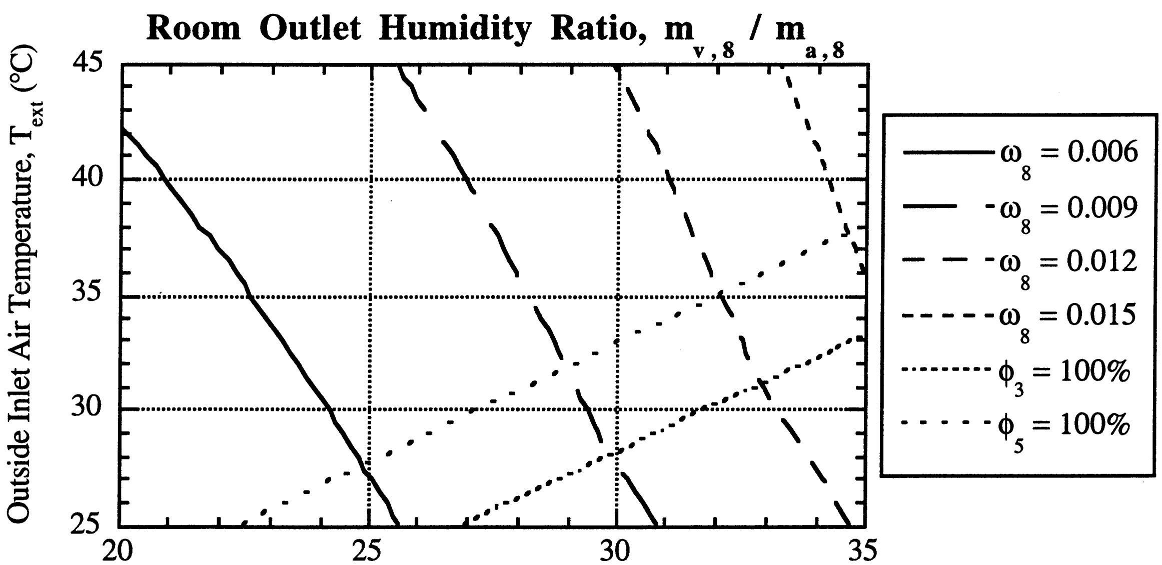
Room Inlet Temperature, T_{int} (°C)
Relative Humidity = 55%
Heat Exchanger Effectiveness = 70%



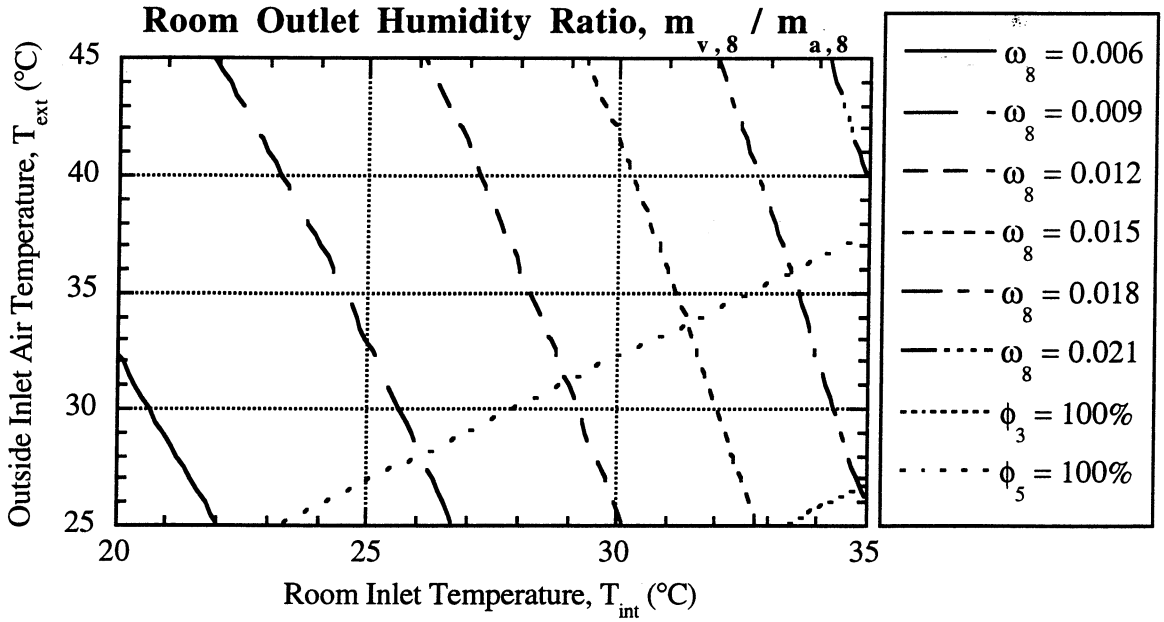
Room Inlet Temperature, T_{int} (°C)
Relative Humidity = 55%
Heat Exchanger Effectiveness = 90%



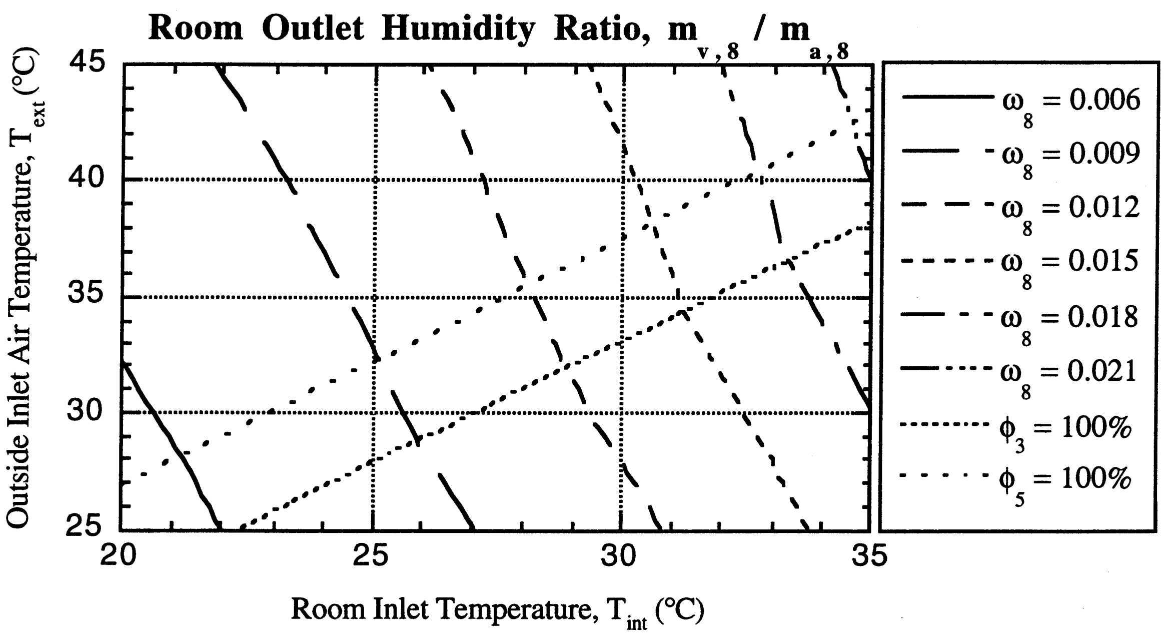
Room Inlet Temperature, T_{int} (°C)
Relative Humidity = 75%
Heat Exchanger Effectiveness = 70%



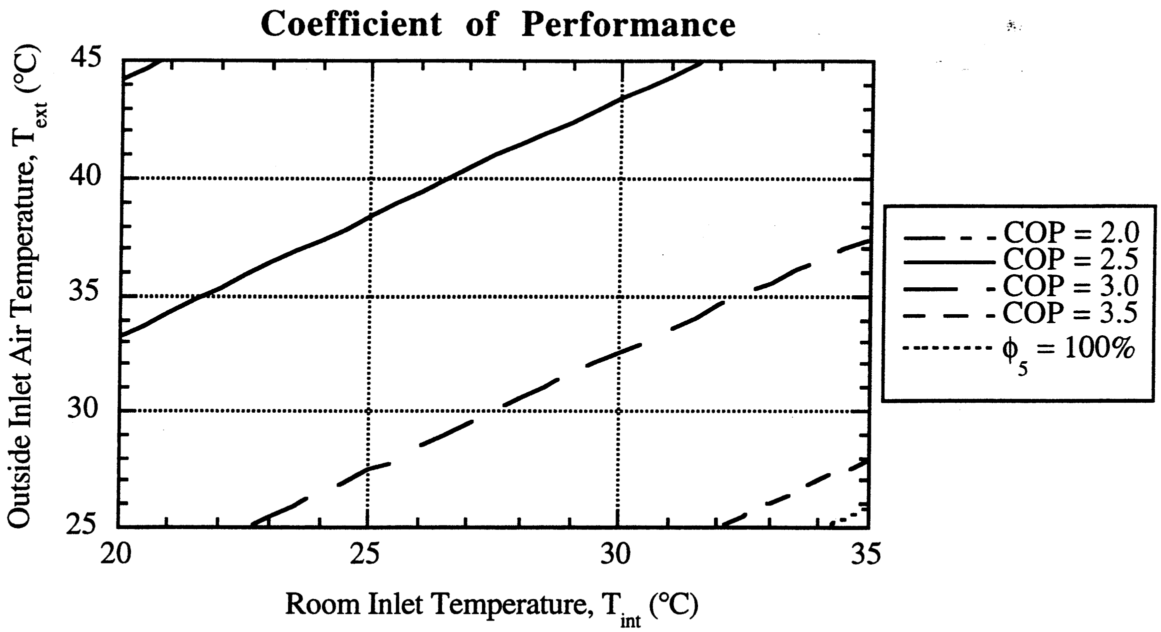
Room Inlet Temperature, T_{int} (°C)
Relative Humidity = 75%
Heat Exchanger Effectiveness = 90%



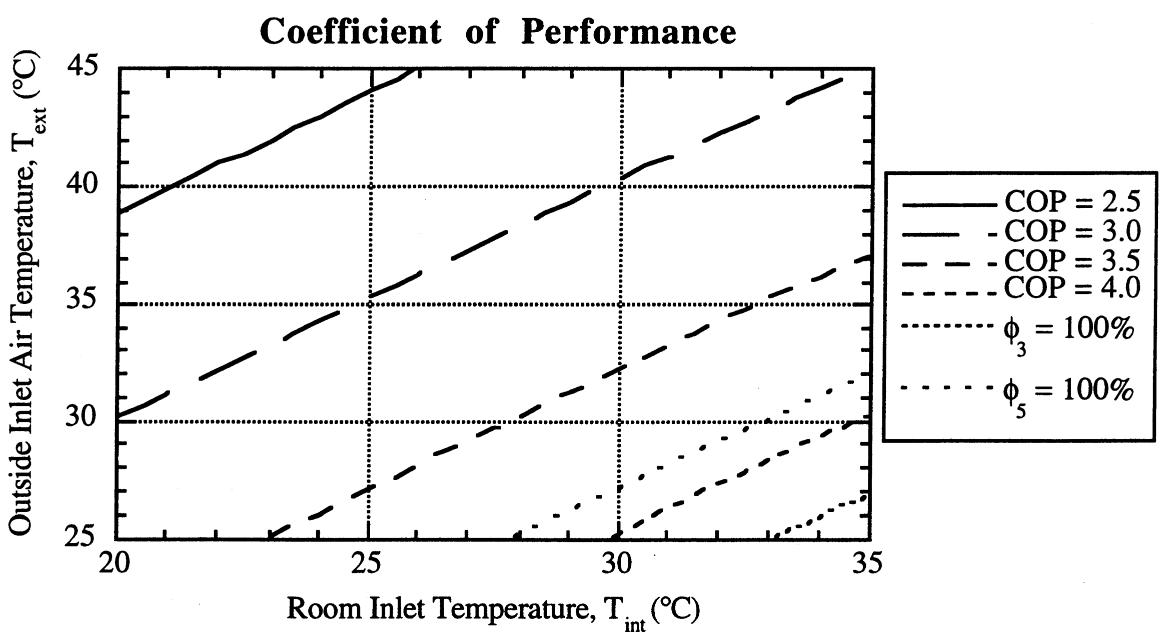
Relative Humidity = 95%
Heat Exchanger Effectiveness = 70%



Relative Humidity = 95%
Heat Exchanger Effectiveness = 90%

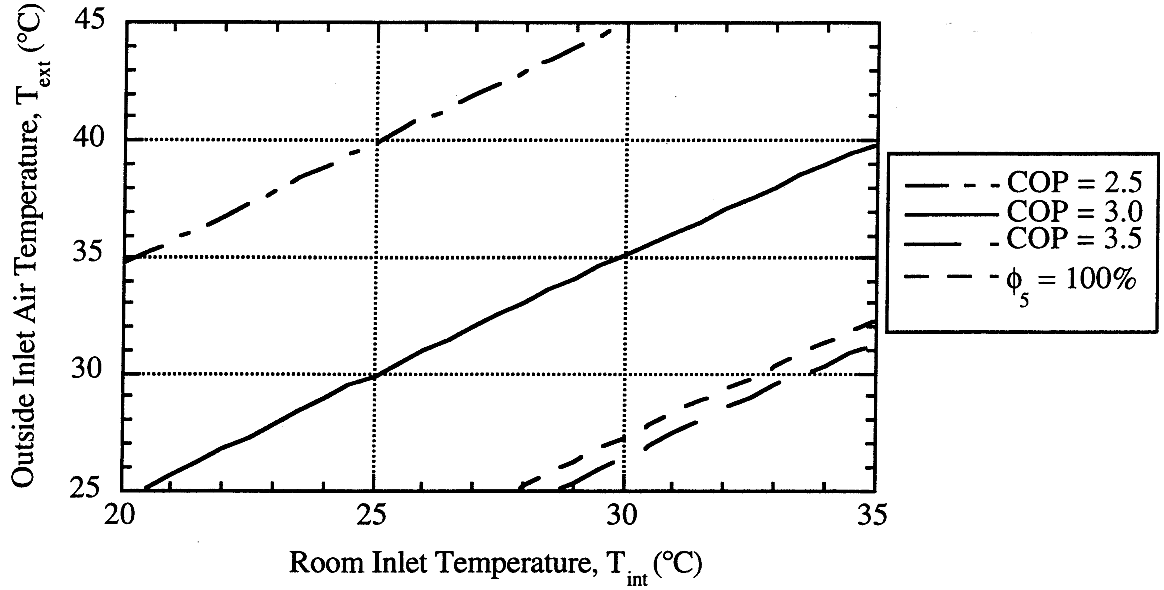


Relative Humidity = 55%
Heat Exchanger Effectiveness = 70%



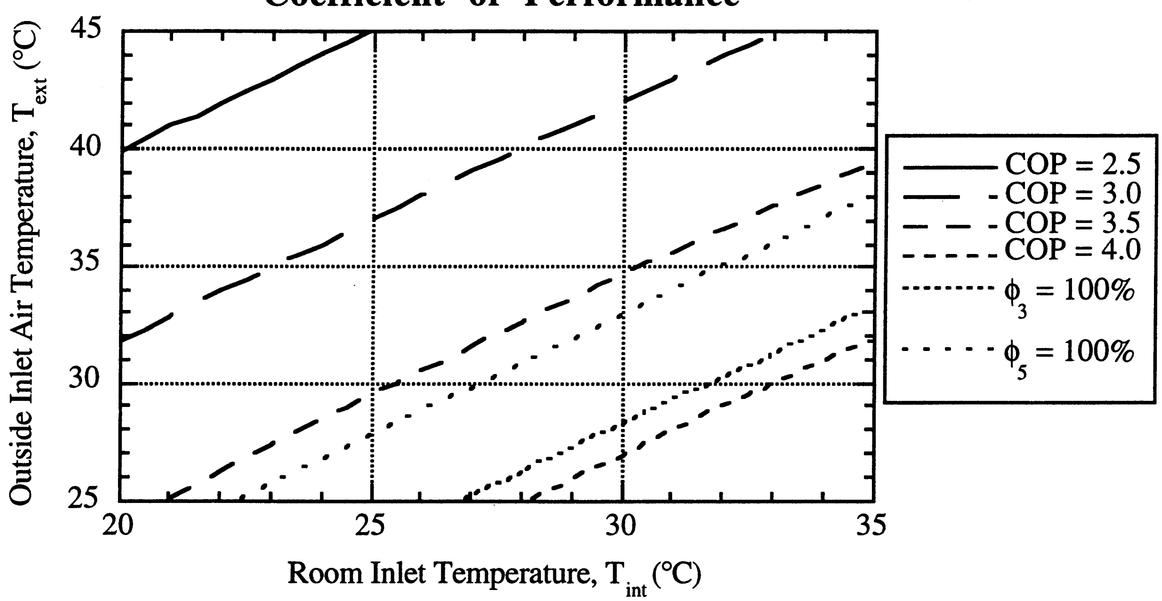
Relative Humidity = 55%
Heat Exchanger Effectiveness = 90%

Coefficient of Performance

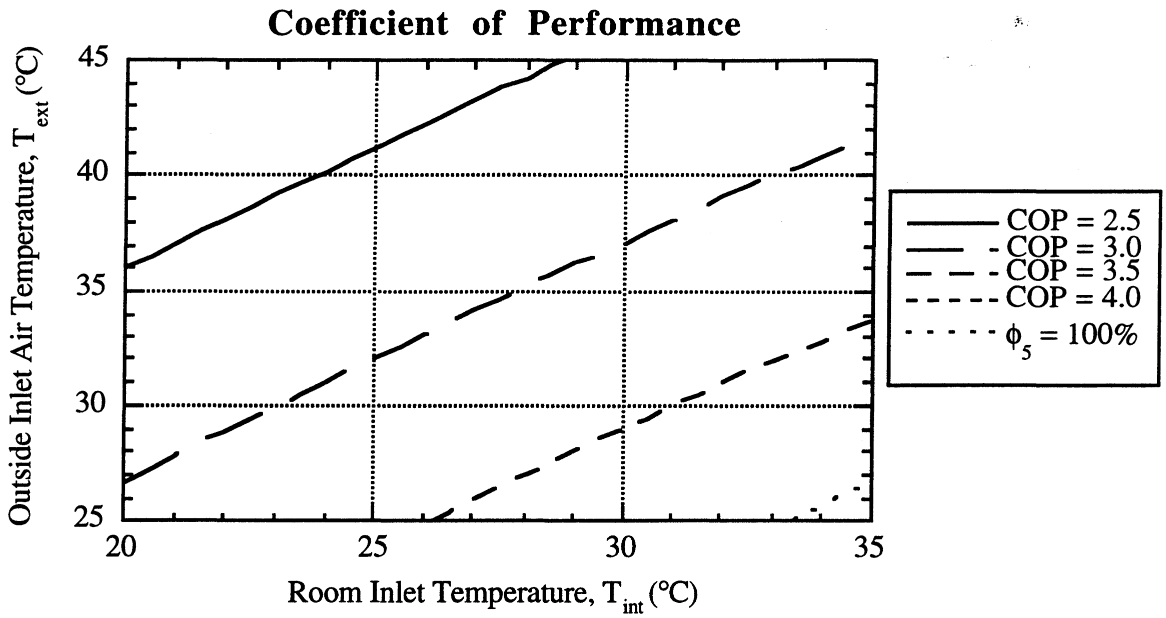


Relative Humidity = 75%
Heat Exchanger Effectiveness = 70%

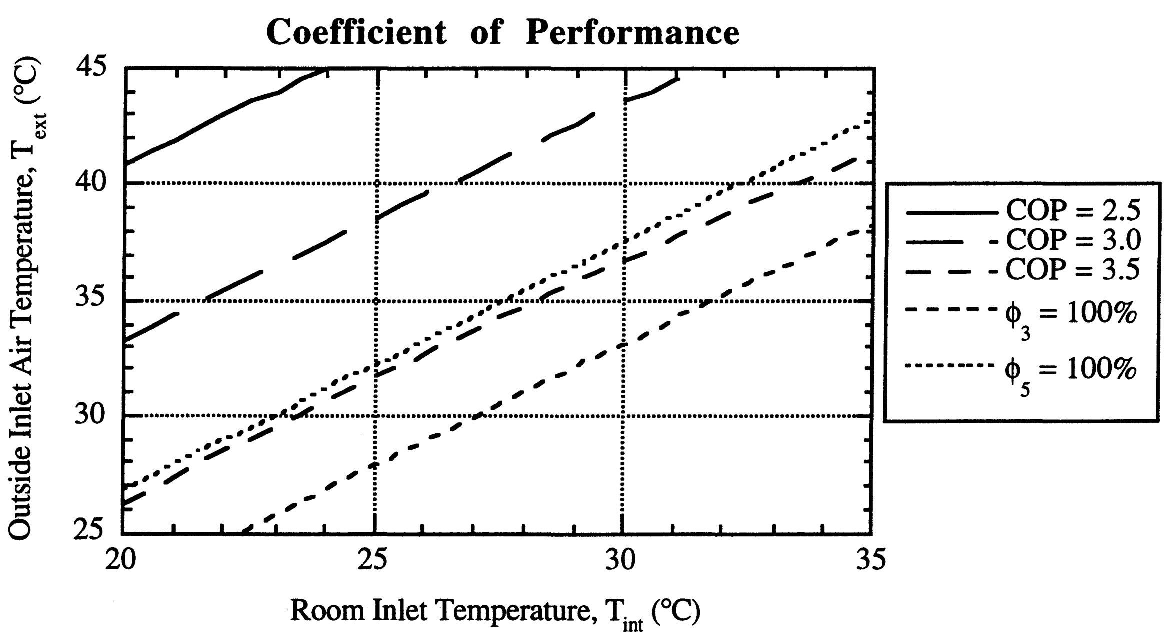
Coefficient of Performance



Relative Humidity = 75%
Heat Exchanger Effectiveness = 90%



Relative Humidity = 95%
Heat Exchanger Effectiveness = 70%



Relative Humidity = 95%
Heat Exchanger Effectiveness = 90%